12-14-2021 10:00 AM

Situation-Aware Quality of Service Enhancement for Heterogeneous Ultra-Dense Wireless IoT Networks

Sabin Bhandari, *The University of Western Ontario*

Supervisor: Wang, Xianbin, *The University of Western Ontario*

A thesis submitted in partial fulfillment of the requirements for the Doctor of Philosophy degree in Electrical and Computer Engineering

Abstract

By engaging a massive number of heterogeneous devices, future Internet of Things (IoT) systems are expected to support diverse applications ranging from eHealthcare to industrial control. In highly-dense deployment scenarios such as Industrial IoT (IIoT) systems, meeting the stringent Quality of Service (QoS) requirements such as low-latency and high reliability becomes challenging due to the uncertainty and dynamics within the IoT networks. To enhance the overall QoS performance, this thesis aims to address the technical challenges of IoT networks. Firstly, to enhance the network reliability, a cloud-assisted priority-based channel access and data aggregation scheme is proposed to minimize the network latency. Besides, the joint impact of packet scheduling and aggregation is considered by using the preemptive M/G/1 queuing model. Subsequently, the sector-based device grouping scheme is proposed for fast and efficient channel access in IEEE 802.11ah based IoT networks. In the proposed framework, the Access Point (AP) forms the sectors and divides into different groups according to the number of stations and their corresponding locations. In addition, the sector-based grouping allows the substantial improvement on packet collision rate and the throughput by utilizing the spatially orthogonal access mechanism.

Similarly, provisioning of accurate synchronization and low latency communication has become critical for IoT networks to support distributed sensing and control. Due to the contention-based channel access, achieving accurate synchronization could be extremely challenging. An efficient clock synchronization scheme is proposed to enhance the synchronization precision of the event critical applications. The proposed scheme assigns time slots with high preference to the timestamp packets of critical nodes and also guarantees the channel access in event-based situations. Furthermore, the proposed scheme provides the deterministic packet scheduling, reduces the channel access delay, and enhances the synchronization precision.

Moreover, in mobile IoT networks such as Unmanned Aerial Vehicle (UAV) networks, mobility of the UAV and the corresponding network dynamics cause frequent network adaptation.

One key challenge caused by this in Flying Ad-hoc Network (FANET) is how to maintain the link stability such that both the packet loss rate and network latency can be reduced. To solve this problem, a location-based *k*-means clustering algorithm is proposed by incorporating the mobility and relative location of the UAVs to enhance the performance and reliability of the UAV network. The principle of the proposed mechanism is to enhance the stability and accuracy of the network by reducing unnecessary overheads and network latency through incorporating several design factors with minimum resource constraints. To further improve the network performance, the CH facilitated optimal collaborative computing scheme is proposed by considering both the computing capabilities and the communication link status. Moreover, the graph-based wireless link scheduling algorithm is present to find the shortest distance to transfer the information among UAVs to deal with a link scheduling problem. Simulation results show that the proposed method significantly reduces the network overheads and improves packet delivery ratio and network latency as compared to the conventional schemes.

Keywords: Internet of Things (IoT), Latency minimization, Cloud-center, Data aggregation, IEEE 802.11ah, Grouping, Random Access, Network Throughput, Time synchronization, MAC protocol, Access Delay, Unmanned aerial vehicles (UAVs), Clustering.

Lay Summary

In dense Internet of Thing (IoT) scenarios such as smart cities and industrial automation systems, providing reliable and low latency communication becomes more challenging due to random access nature of the underlying wireless IoT networks. The uncoordinated transmissions of data packets by the densely spread sensors could cause high latency and packet collisions. To overcome the above-mentioned issues, a cloud-assisted priority-based data aggregation and scheduling scheme is proposed to minimize the network latency. Subsequently, to further enhance the performance of a network, a device grouping mechanism is proposed to provide the efficient exchange of information among the network entities. The main aspects of this technique include how to organize groups, how to allocate access slots to individual groups to reduce the packet collisions, and to improve the overall efficiency.

Clock synchronization is another crucial issue in IoT applications to perform the event-driven measurement, which often requires accurate common reference time for collaborative information exchange. The critical challenge for dense IoT communication lies in facilitating the synchronized channel access to the large number of devices by supporting the unique traffic characteristics. In this regard, the clock synchronization algorithm based upon the Medium Access Control (MAC) layer time stamping is proposed to minimize the global and local synchronization error with low message complexity and scalability.

In Unmanned Aerial Vehicle (UAV) networks, random mobility of a UAV causes increased network dynamics as well as energy and bandwidth consumption. A key challenge is to maintain the link stability between the nodes to minimize the packet drop rate and network latency. To solve this problem, a stable clustering scheme is proposed for randomly deployed UAV networks by incorporating the mobility and coverage probability. Besides, the cluster maintenance scheme is also presented with reference to the relative mobility and locations to enhance the stability of the cluster network. Finally, to address the issues regarding limited energy and computational resources, the collaborative-computing scheme is designed to minimize the total energy consumption of the UAV nodes, while satisfying the latency constraint in the given network conditions.

To those whom I love

Acknowledgments

First and foremost, I would like to express my heartfelt gratitude and sincere appreciation to my supervisor Dr. Xianbin Wang for his faith in me and bestowing a platform to bring the best out of me. His persistent guidance, invaluable advices, and continuous encouragement in all possible ways have helped immeasurably in pushing me further and achieving the success of this level. Without his mentoring and support, I could not have completed this dissertation.

Besides my supervisor, I am also grateful to my thesis examiners Dr. Lei Lei, Dr. Abdallah Shami, Dr. Weiming Shen, and Dr. Evgueni Bordatchev for their constructive comments and invigorating suggestions. All their insights regarding the research work have helped me to improve the quality of this dissertation. I would also like to extend my sincere thanks to Dr. Shree Krishna Sharma for his feedback and suggestions during my research.

My sincere acknowledgement to the Department of Electrical and Computer Engineering for such a wonderful opportunity and an atmosphere that has helped me grow from strength to strength in so many ways, academically, and otherwise.

Last but not the least, I cannot stop myself from thanking all present and former lab members of our research group for their moral support and standing by me in all odds and evens. My fellow lab members have always been motivating and generous enough, both in and out of the lab.

Most importantly, I am immensely indebted to my Fupu, my wife, and other family members for their unequivical support during my Ph.D. journey. My wife always supported me with her unconditional love and care. I am therefore grateful to them and would like to dedicate this thesis to my family.

Contents

List of Figures

List of Tables

List of Abbreviations

5G	Fifth Generation
AC-BC	Ant Colony-Bee Colony
ACK	Acknowledgment
ACO	Ant Colony Optimization
AI	Artificial Intelligence
AID	Association ID
AODV	Ad-hoc On-Demand Distance Vector
AP	Access Point
AR	Augmented Reality
BIMPC	Bio-Inspired Mobility Prediction Clustering
BS	Base Station
CAP	Contention Access Period
CFP	Contention Free Period
CH	Cluster Head
CSI	Channel State Information
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTS	Clear-To-Send
D2D	Device-to-Device
DBSCAN	Density-based Spatial Clustering Applications with Noise
DCF	Distributed Coordination Function
DIFS	DCF Interframe Space
EALC	Energy-Aware Link-based Clustering
ED	Event Driven
EDD	Earliest Due Date
EI	Emergency Indication
EPCAP	Explicit Prioritized Channel Access Protocol
FANET	Flying Ad-hoc Network

List of Abbreviations (Cont'd)

FCFS	First Come First Served
FIFO	First-In-First-Out
FTSP	Flood Time Synchronization Protocol
GPS	Global Position System
GS	Ground Station
GTS	Guaranteed Time Slots
GWO	Grey Wolf Optimization
IoE	Internet of Everything
IoT	Internet of Things
IIoT	Industrial IoT
LoS	Line-of-Sight
LQP	Link Quality Prediction
M2M	Machine-to-Machine
MAC	Medium Access Control
MANETs	Mobile Ad hoc Networks
MCC	Mobile Cloud Computing
MEC	Mobile Edge Computing
MIMO	Multiple Input Multiple Output
ML	Machine Learning
MLSC	Mobility and Location-aware Stable Clustering
MPCA	Mobility Predication Clustering Algorithm
MST	Minimum Spanning Tree
ND	Normal Data
NLoS	Non-Line-of-Sight
NTP	Network Time Protocol
OLSR	Optimized Link State Routing
PBS	Pair-wise Broadcast Synchronization
PDR	Packed Delivery Ratio

List of Abbreviations (Cont'd)

PHY	Physical
PRR	Packet Received Rate
PSO	Particle Swarm Optimization
PSP	Packet Schedule Phase
PTP	Packet Transmission Phase
QoS	Quality of Service
RAW	Random Access Window
RBS	Reference Broadcast Synchronization
ROS	Receiver Only Synchronization
RSP	Ratio-based time Synchronization Protocol
RSSI	Received Signal Strength Indicator
RTS	Request-To-Send
SINR	Signal to Interference plus Noise Ratio
SRS	Sender-Receiver Synchronization
TDMA	Time Division Multiple Access
TGah	Task Group ah
TIM	Traffic Indication Map
TPSN	Time-sync Protocol for Wireless Sensor Network
UAV	Unmanned Aerial Vehicles
ULSNs	UAV-based Linear Sensor Networks
URP	UAV Routing Protocol
UTC	Coordinated Universal Time
VANETs	Vehicular Ad hoc Networks
WLAN	Wireless Local Area Network
WSNs	Wireless Sensor Networks

Chapter 1

Introduction

1.1 Overview of Ultra-dense IoT Networks

With the rapid advancements in digital electronics and wireless communications, the application areas of Internet of Things (IoT) have increased significantly and they support a wide range of applications including industrial automation, intelligent transportation, medical, and eHealthcare services [1]. In addition, IoT is also considered as an integral part of future Internet and comprises with the millions of intelligent communicating objects or *things*. The IoT term is also referred as the Internet of Everything (IoE) and it basically connects the people, *'things'*, processes, and data together to fulfill the everyday needs of society in an effective way [2]. The emerging Wireless Sensor Networks (WSNs), Device-to-Device (D2D), and Machine-to-Machine (M2M) technologies have significant impact to extend the sensory capabilities of different sensors, thus enabling the concept of wireless IoT [3]. The widely used *Intranet of Things* [4] usually refers to local networks with the set of paradigms such as M2M, D2D, and WSN, and only have the regional information. However, IoT networks can exploit comprehensive and historical information by collaborating with different intranets and the cloud server.

The emerging IoT paradigm is expected to interconnect various objects and processes for massive information collection, analysis and utilization [5]. Many industries are putting significant efforts in creating novel business models, products and services based on IoT platforms towards bringing economic and social benefits in various sectors such as industrial automation, health care, and transportation [6]. Consequently, a total number of connected sensors and machine type communication devices has been rapidly increasing over the recent years, and IHS Markit has predicted about 125 billion connected devices by 2030. However, due to limited radio resources available to support these massive number of connections and cost issues, the upcoming fifth generation (5G) wireless networks are expected to support extremely high device density up to about 1 million devices per square kilometers [7], thus leading to ultra-dense wireless IoT networks.

An overview of different challenges in ultra-dense IoT networks and the main contribution of the thesis are presented in section 1.2 and section 1.3, respectively.

1.2 Challenges in Ultra-dense IoT Networks

In ultra-dense IoT application scenarios such as industrial automation/control systems, providing fast channel access, and reliable and low-latency communication links becomes extremely challenging due to inefficient channel access mechanism, resource-constrained edge devices and limited available radio resources [8]. Due to contention-based nature of the most existing channel access schemes in unlicensed wireless networks, the problem of access network congestion becomes severe in ultra-dense IoT networks since the collision rate increases dramatically with the device density. Although the traditional IEEE 802.11 standard works well for the small local wireless network with a single Access Point (AP) supporting a reasonably small number of devices [9], scalability becomes the main issue in ultra-dense networks due to significant increase in the channel access delay and packet collision rate. Another major problem in ultra-dense networks comes from the device heterogeneity since the network has to support diverse Quality of Service (QoS) requirements of various IoT services. In the following subsections, the above-mentioned challenges are described in detail.

1.2.1 System Delay and Medium Access Control Protocol

In an IoT network, the overall system latency depends on a set of parameters such as distance and a number of hops towards the destination node, data rate, node density, Medium Access Control (MAC) and routing protocols, and the available energy and computational resources at the nodes. All the above-mentioned parameters may lead to unpredictable and high end-to-end latency. Out of these, the employed MAC layer protocol determines the one-hop delay and the network layer is responsible for controlling the multi-hop delay. The one-hop delay τ_{hd} resulted at the MAC layer can be expressed in terms of different delay components as follows [10]

$$\tau_{hd} = \tau_{prd} + \tau_{qd} + \tau_{cd} + \tau_{td} + \tau_{pgd} + \tau_{rd}, \tag{1.1}$$

where τ_{td}, τ_{rd}, and τ_{pgd} denote transmission, reception, and propagation delays, respectively, and they are hardware dependent. Similarly, τ_{prd}, τ_{cd}, and τ_{qd} are the processing, channel access, and queuing delays, respectively, and higher latency may result due to the queuing of the packets and the time required to access a channel. The critical MAC layer challenge for IoT networks is to facilitate the channel access to an extremely large number of devices with unique traffic characteristic and diverse service requirements. To enhance the overall performance, this thesis aims to focus on improving queuing strategies and channel access techniques to ensure QoS requirements.

1.2.2 Prioritized Channel Access Mechanism and Synchronization Precision

In event critical wireless IoT network, a high level of synchronization precision is required to reduce the idle listening times and ensure reliable delivery of the sensed information. Also, the network reliability depends so much on the probability of packet loss due to the congestions, collisions, and link constraints and decreases as the number of nodes increases in the network [4]. However, the implication of priority-based deterministic channel access mechanism contributes significantly to reduce the access latency, packet congestion, and packet collisions of critical nodes. Consequently, the reliability of critical nodes gets enhanced as compared to the normal nodes [27]. The proposed scheme prioritizes the most urgent traffic for channel access to complete its transmission within a deadline bound to provide guaranteed channel access in emergency and event-based situations, where multiple sensor nodes are triggered simultaneously to transmit time-critical data to the coordinator. Besides, the proposed scheme assigns time slots with high preference to the timestamp packets of critical nodes by using different MAC layer attributes and guarantees the channel access in event-based situation.

1.2.3 Device Heterogeneity

Another major problem in ultra-dense networks comes from the device heterogeneity since the network has to support diverse QoS requirements of various IoT services. One of the promising approaches to address this issue is to employ a suitable device grouping mechanism by enabling the efficient exchange of information among the network entities. This thesis also focuses on improving the efficiency of channel utilization in terms of system delay and throughput by employing a device grouping mechanism. The sector-based grouping scheme allows the substantial improvement on packet collision rate/probability and the throughput of ultra-dense IoT networks.

1.2.4 Mobility and Dynamic Topology

The mobilities of the nodes are application dependent. In the case of Unmanned Aerial Vehicles (UAV), the mobility of nodes is higher than that of Vehicular Ad hoc Networks (VANETs) and Mobile Ad hoc Networks (MANETs) [11]. The UAV nodes are highly mobile, with the speeds of 30 to 460 km/h [12]. The node mobility causes the significant impact on the network performance of mobile networks. The peer-to-peer connections are formed among the mobile nodes to maintain the coordination and collaboration and it can be effectively achieved by clustering /grouping [13]. For the homogeneous small-scale IoT network, a single grouping is a best choice; however, for multi-purpose heterogeneous networks, there is a need for multi-cluster network. In this scenario, the Cluster Head (CH) is responsible for the down-link communication and inter-cluster communication. In the process of clustering, after forming the dynamic clusters, the mobile nodes are relocated at the positions vertically projected on the centroid of clusters.

Network formation is tightly coupled with the formation of multi-mobile networks. To manage the large number of mobile nodes and several static ground stations is one of the significant challenges. An extensive set of mini networks can be formed as an intelligent swarm. The self-organized network formation is an example of the intelligent cluster formation, where the nodes are self-organize to reconnect themselves after a disruption in connections [14].

1.2.5 Variable Communication Links

In mobile IoT networks, the network may have different types of communication links such as UAV-to-UAV and UAV-to-ground link. The key features of mobile networks are reliability and survivability through redundancy. The IEEE 802.11 standard is widely used in mobile networks [15]. In the case of UAV networks, the IEEE 802.15.4 standard can be effectively used for UAV-to-UAV communications. The IEEE 802.15.4 enables a low power and less complicated implementation with lower data rate [16]. For the UAV-to-ground commutations, the IEEE 802.11 can be used due to the high data rates and long-range coverage. Moreover, for the real time communication, the MAC layer should address the challenges such as packet delays, mobility, variable link quality, and optimal channel utilization.

1.3 Thesis Contributions and Outline

By considering the challenges mentioned above, technical issues on ultra-dense IoT system are addressed in this thesis. The main contributions of this thesis are as follows:

In Chapter 2, a priority-based channel access and data aggregation scheme at the CH to reduce channel access and queuing delays in a clustered industrial IoT network is proposed.

First, a prioritized channel access mechanism is developed by assigning different MAC layer attributes to the packets coming from two types of IoT nodes, namely, high-priority and low-priority nodes, based on the application-specific information provided from the cloud-center. Subsequently, a preemptive M/G/1 queuing model is employed by using separate low-priority and high-priority queues before sending aggregated data to the cloud server. The simulation results show that the proposed priority-based method significantly improves the system latency and reliability as compared to the non-prioritized scheme.

In Chapter 3, a sector-based device grouping scheme is proposed for fast and efficient channel access in IEEE 802.11ah based IoT networks such that a total number of the connected devices within each sector is dramatically reduced. In the proposed framework, the AP divides its coverage area into different sectors, and then each sector is further divided into distinct groups based on a number of the devices and their location information available from the cloud-center. Subsequently, individual groups within a sector are assigned to specific Random Access Window (RAW) slots, and the devices within the distinct groups in different sectors access the allocated RAW slots by employing a spatial orthogonal access mechanism. The performance of the proposed sectorized device grouping scheme has been analyzed in terms of system delay and network throughput. The simulation results show that the proposed scheme can significantly enhance the network throughput while simultaneously decreasing the system delay as compared to the conventional Distributed Coordination Function (DCF) and IEEE 802.11ah grouping scheme.

In Chapter 4, an efficient MAC protocol for supporting distributed synchronization through guaranteed channel access for time-critical sensor nodes is proposed. A priority-based fast and efficient channel access scheme proposed in Chapter 2 is utilized to enhance the synchronization precision of the event-critical IoT nodes. The proposed protocol assigns time slots to timestamp packets of the time-critical nodes using a prioritized channel access mechanism, and also guarantees channel access in event-based situations. In addition, the proposed protocol also provides a deterministic scheduling for the scenarios where the delay bound of a certain priority traffic changes based on the circumstances of the emergency situation. The simulation results show that the proposed scheme significantly improves the synchronization precision of the event critical sensor nodes, and also enhances the reliability of overall IoT networks.

In Chapter 5, a location-based *k*-means clustering algorithm by incorporating the mobility and relative location of the UAVs is proposed to enhance the performance and reliability of the UAV network with limited resources. The objective of the proposed Mobility and Location-aware Stable Clustering (MLSC) mechanism is to enhance the stability and accuracy of the network by reducing unnecessary overheads and network latency through incorporating several design factors with minimum resource constraints. In addition, the relationship between the maximum coverage probability of CH and cluster size is derived to find the optimal cluster size to minimize the network overhead. The cluster maintenance mechanism is also presented with reference to the relative mobility and locations to enhance the stability of the cluster network. Moreover, the graph-based link scheduling algorithm is proposed to find the shortest distance to transfer the information among the UAVs to overcome the latency and link schedule problems. The simulation results show that the proposed MLSC scheme significantly reduces the network overheads, and also improves packet delivery ratio and network latency as compared to the conventional clustering methods.

In Chapter 6, the distributed computing framework for UAV networks is proposed to share a common Access Point (AP)/CH to perform a set of tasks collaboratively. The tasks are optimally distributed among the UAV nodes to minimize the total energy consumption by satisfying the latency constraint. Firstly, the low complexity solution to determine the optimal locations of CH UAVs is presented to enhance the network capacity by using *k*-means clustering and their altitude is determined to optimize the QoS metrics. Besides, a graph-based link scheduling method proposed in Chapter 5 is utilized to find the shortest path among the UAV nodes to transfer the information to reduce the overall communication latency. Finally, the optimal collaborative computing scheme is derived by considering both the computing capabilities of the UAV nodes and the communication link status. The simulation results illustrate the benefits of the proposed collaborative computing scheme over the conventional schemes in terms of both energy and latency constraints.

Finally, all the contributions are summarized in Chapter 7 with the identification of future research directions.

Chapter 2

Latency Minimization Using Prioritized Channel Access and Data Aggregation

2.1 Introduction

Internet of Things (IoT) has been emerged as a new paradigm that interconnects various objects and processes for distributed real-time information collection and utilization in several applications [17]. A typical IoT architecture mainly consists of four interconnected sub-systems, including connected intelligent objects/things through a sensor network, routers/gateways at the edge, backbone communication infrastructure, and the clouds [18]. Today's developments in the Wireless Sensor Networks (WSNs), Device-to-Device (D2D), Internet, Machine-to-Machine (M2M), and mobile computing technologies have a significant impact to extend the sensory capabilities of IoT networks [19]. However, due to the large-scale and highly-dense nature of many IoT applications, performing timely acquisition and analysis of IoT related data is crucial to support low-latency applications.

Among many potential applications, industrial IoT is considered as a key enabler for industrial automation, intelligent transportation, logistics and control systems [6]. Various application requirements have brought many challenges to design more efficient and reliable industrial IoT networks. The main challenges in industrial IoT networks include low latency, low per node energy consumption, reliability, and secure data transmissions to the application servers [2]. Out of these, IoT network latency has been considered as one of the most critical issues in industrial automation and control sub-systems. The main network parameters that affect the system delay are node density, data rate, and energy per node, processing power, routing protocol, and Medium Access Control (MAC) protocol [20]. To deal with the latency issue, an IoT network must be designed to meet the real-time requirements of the aforementioned application scenarios [2]. One of the potential approaches to reduce system delay in dense wireless IoT networks is to devise a suitable MAC protocol, which can effectively regulate the access of limited channel resources. At the MAC layer, several factors such as overhearing, over-emitting, collisions, and control packets overhead affect the overall system delay. These factors are generally related to the radio operating mode, the medium access technique and the service time.

In this Chapter, a cloud-assisted priority-based channel access and data aggregation scheme is proposed for irregularly deployed sensor nodes to minimize the network latency and to enhance the system reliability of IoT networks. The cloud center is equipped with massive processing power, and storage capabilities [21], however, it does not support low-latency applications [22, 23]. In the considered framework, the Cluster Head (CH) extends the cloud's functions to the edge of the network by prioritizing and aggregating the incoming data packets, and the cloud-center provides various levels of information such as priority levels and locations of the IoT nodes to the CH. A priority based channel access scheme is employed at the CH to reduce the channel access latency by assigning different MAC layer attributes to the incoming data packets. Subsequently, the prioritized data packets are sent to the separate queues according to their priority levels and are aggregated before sending to the cloud via a gateway. Finally, the performance of the proposed joint prioritized channel access and data aggregation is analyzed using the preemptive M/G/1 queuing model and compared with the conventional non-prioritized scheme.

The rest of this chapter is organized as follows. Section 2.2 summarizes the related works. In Section 2.3, the overall system model of a hierarchical IoT network and proposed prioritized channel access and data aggregation schemes are described in detail. In Section 2.4, the performance of the proposed method is evaluated via simulations. Finally, the chapter is concluded in Section 2.5.

2.2 Related Works

A number of MAC protocols have been proposed based on the IEEE 802.15.4 standard [24] to address the latency issues. A MAC protocol based backoff time decision rule has been presented in [25] for a hierarchical Machine-to-Machine (M2M) network having different clustered nodes. Besides, a mathematical model has been introduced in [26] for superframe and access latency of the MAC protocol for an industrial IoT environment based on the queuing theory. Furthermore, an extended channel access mechanism namely, Explicit Prioritized Channel Access Protocol (EPCAP) [27] has been studied based on the IEEE 802.15.4 standard. The EPCAP proposed in [27] incorporates different traffic priority levels to handle critical events and utilizes M/G/c based multi-server queuing network system.

The level of network latency can be further reduced by dividing incoming data packets into different queues, and subsequently by employing a suitable data aggregation scheme. The data aggregation process helps to eliminate the data redundancy, to minimize the communication load, and hence to reduce the overall network latency. In this regard, the authors in [28] proposed to employ a data aggregation scheme to reduce the network signaling load. In addition, a tunnel based data aggregation method has been proposed in [29], in which an aggregator merges the M2M data packets, appends with its own packet, and forwards the aggregated data to the gateway/base station. Besides, the authors in [30] proposed a priority based data aggregation scheme for M2M communication over the cellular network to maintain the trade-off between delay requirements and power constraints by using a preemptive M/G/1 queuing model. However, the existing works did not consider the joint impact of priority based channel access and prioritized queuing in heterogeneous IoT networks. In addition, the potential benefit of involving the cloud in latency reduction at the IoT edge network has not been considered.

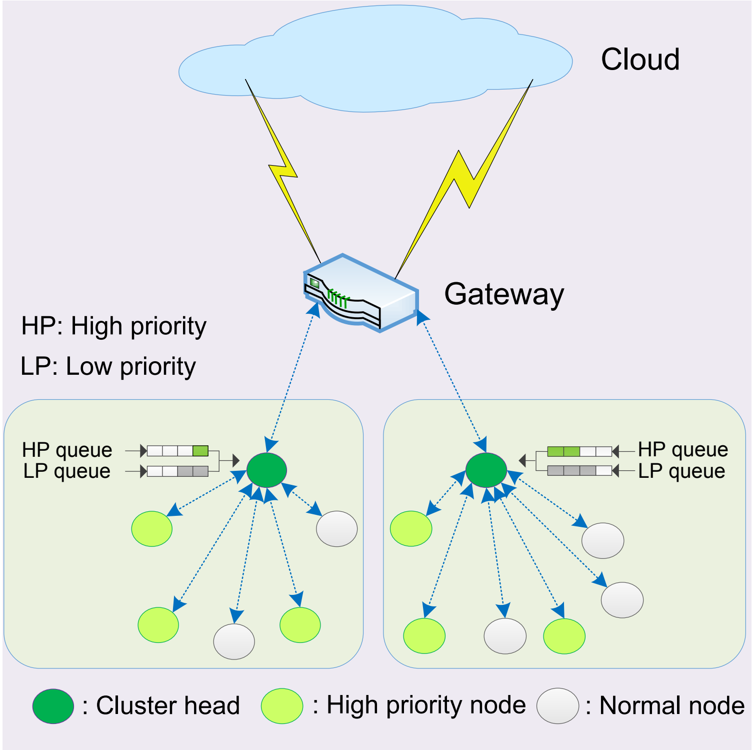

Figure 2.1: System model for the hierarchical IoT network.

2.3 System Model and Proposed Schemes

An industrial IoT scenario, composed of N number of heterogeneous sensor nodes deployed over an area of $l \times l$ m^2 (i.e., rectangular industrial sub-unit) is considered as shown in Fig. 2.1. In each industrial unit, the data gathered by N sensors are classify into two classes, i.e., normal data (ND) and event driven (ED) data traffic. The ND packets are regularly generated by low priority P_l nodes during some process-related measurements, while the sporadic ED packets are triggered by high priority P_h nodes when a physical quantity detected by a sensor crosses its threshold. Each node supports only one type of data, i.e., either ND or ED. Also, M out of N nodes transmit high priority packets, i.e., P_h packets and the remaining nodes transmit only low priority packets, i.e., P_l packets. In addition, the proposed network topology is considered to be static over the time. The gateway and the cloud-center are connected via high-speed wireless links with negligible latency and packet loss.

All the deployed sensor nodes are associated with the CH/aggregator. Also, we consider that the nodes including CH and the gateway have the child-parent relationships. All the sensor nodes belonging to the same CH contend to access the channel to the corresponding parent node of the link. Data generated from terminal nodes are transmitted to the gateway after data aggregation at the CH for subsequent transmission to the application server. The gateway and CH are considered to be positioned at the specific locations and usually have the higher energy and computational power as compared to the sensor nodes. The CH can get the application-specific information such as priority levels and locations from the cloud application server. In the considered system setup, the queuing delay for each priority class depends on the scheduling policy adopted at the CH.

2.3.1 Prioritized Channel Access Mechanism

The data prioritization and delay modeling are performed by the application layer by considering the MAC layer parameters according to the requirements of industrial applications and the network conditions. The IEEE 802.15.4 standard uses Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) to access the radio channel. However, CSMA/CA is not suitable for the delay critical industrial applications since it does not include the prioritization and delay responsiveness properties [31]. In the industrial IoT systems, flow control, process monitoring, and fault detection sub-system must have priority and delay aware medium access mechanisms.

Figure 2.2 shows the timing diagram of different nodes contending the channel access according to their priority levels. In this scenario, any packets in the low priority queue will not be served until the high priority queue becomes empty. The P_h nodes always have the fixed short backoff period, more frequent Common Channel Access (CCA) detection, and high number of backoffs. However, the P_l nodes use longer random backoff period, less frequent detection, and lower number of backoffs. In addition, CCA detection time of P_l nodes is considered to be longer than the sum of CCA detection time and the backoff periods of P_h nodes.

The behavior of the CSMA/CA is affected by different MAC parameters such as the minimum backoff exponent (*macMinBE*), the maximum backoff exponent (*macMaxBE*), the intial

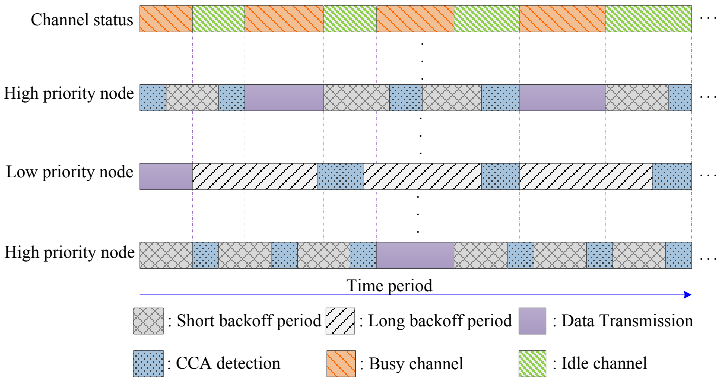

Figure 2.2: Prioritized channel access mechanism.

value of the contention window (*CW*), and the maximum number of backoffs (*macMaxCSMABackoffs*). Different values of these MAC parameters have a great impact on the performance of an IoT network. Instead of having the same value of CSMA/CA parameters for both traffic (i.e., low priority and high priority), we can assign its own attributes for each class. Let us define [$macMinBE_h$, $macMaxBE_h$] and CW_h as the backkoff interval and contention window values for high priority nodes, and [$macMinBE_l$, $macMaxBE_l$] and CW_l as the corresponding values for the low priority nodes. Moreover, by specifying different CSMA/CA parameters, the priority based scheduling can be implemented to reduce the channel access latency of the high priority packets as depicted in Fig. 2.3 [31].

2.3.2 Data Aggregation without Prioritization

In case of the data aggregation without prioritization, the data packets from sensor nodes arrive at the CH and are placed in the queue. The individual packets at the CH are served in different time lengths. The sensor nodes release their data slots and randomly acquire new ones in the next frame, after successfully transmitting a packet in the data transmission period of the current frame. This assumption guarantees that the service times of successive packets are independent and identically distributed (i.i.d.) random variables. Based on this assumption, the queue of each node in the traffic can be modeled as an M/G/1 queuing system [32]. In the

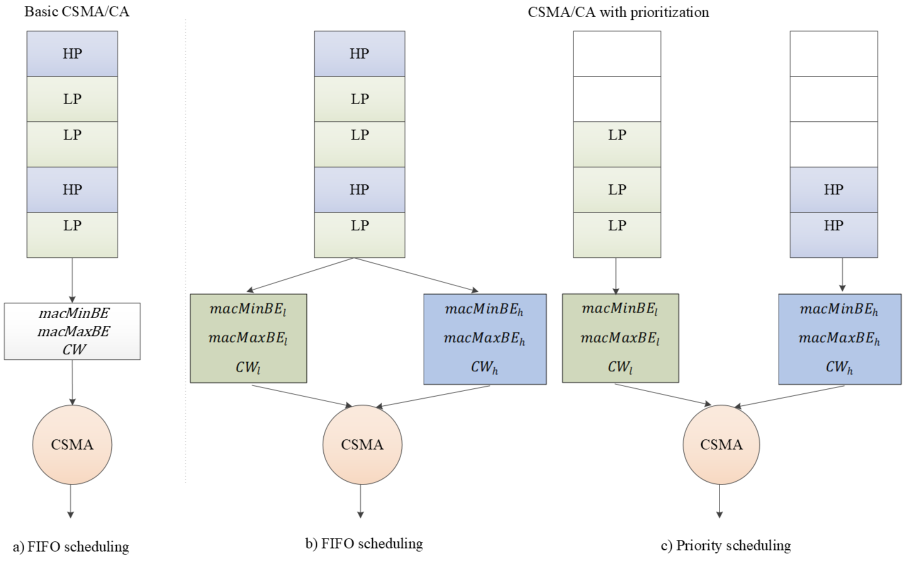

Figure 2.3: Illustrations of the basic CSMA/CA as First-In-First-Out (FIFO) scheduling and the prioritized CSMA/CA.

considered M/G/1 queuing model, the data arrival pattern follows the Poisson distribution with a packet arrival rate λ, and the utilization rate of the packet at the CH is given by

$$\rho = \lambda E[S], \tag{2.1}$$

where $E[S]$ is the expected service time of the aggregated data without priority. The expected waiting time $E[W]$ of the non-priority agrregated data before being served and the expected system delay $E[D_{sys}]$, i.e., the total time that the packet should be in the queue until being transmitted as an aggregated data is given by Pollaczek-Khinchin (P-K) formula and expressed as [28, 33]

$$E[W] = \frac{\lambda E[S^2]}{2(1-\rho)}, \tag{2.2}$$

$$E[D_{sys}] = E[S] + E[W], \tag{2.3}$$

where $E[S^2]$ is the second order moment of the service time, and can be computed as follows

$$E[S^2] = \frac{4}{3}E[S]^2. \tag{2.4}$$

In this work, the general mathematical model of CSMA/CA procedure of IEEE 802.15.4 presented in [34, 35] is adapted. Using this model, the expected service time can be expressed as [34]

$$E[S] = E[D] + T_{Tx} + 2T_{turn} + T_{ACK}, \tag{2.5}$$

where $E[D]$ denotes the time duration from the epoch that the data packet just arrives at the head of queue to the epoch just before packet transmission or discarded. The T_{Tx} and T_{ACK} are the transmission time of data and acknowledgment (ACK) packet respectively, and T_{turn} is the turnaround time. The parameter $E[D]$ depends on the CSMA/CA procedure and is affected by different MAC parametes such as CW, $macMaxBE$, $macMinBE$, and $macMaxCSMABackoffs$. The expected value $E[D]$ can be expressed as [34]

$$E[D] = \sum_{v=0}^{m} \alpha^{v}(1-\alpha)\left\{\sum_{i=0}^{v} \frac{CW_i - 1}{2}\sigma + (v+1)T_{CCA}\right\} + \alpha^{m+1}\left\{\sum_{i=0}^{m} \frac{CW_i - 1}{2}\sigma + (m+1)T_{CCA}\right\}, \tag{2.6}$$

where T_{CCA} is the time interval for performing CCA, α is the busy channel probability, and σ is the length of backoff slot. The contention window size for the ith retry is given by; $CW_i = min\left\{2^i macMinBE, macMaxBE\right\}$. The default values of $macMinBE$ and $macMaxBE$ are 3 and 5, respectively [34]. The data packets are discarded or dropped after $m + 1$ attempts at CCA, and subsequently the data packet loss rate is given by [34]

$$P_{loss} = \alpha^{m+1}. \tag{2.7}$$

Then, the probability of channel being busy α can be expressed in term of P_{loss} as [34]

$$\alpha = \frac{(N-1)(1-P_{loss})E[N_\tau](T_{CCA}+T_{Tx}+2T_{turn}+T_{ACK})}{\frac{1}{\lambda}+E[N_\tau]E[D]}, \tag{2.8}$$

where N is the number of sensor nodes associated with the CH, N_τ is the number of packets served in a busy period of the M/G/1 queuing system, and $E[N_\tau]=\frac{1}{1-\rho}$. Therefore, by solving

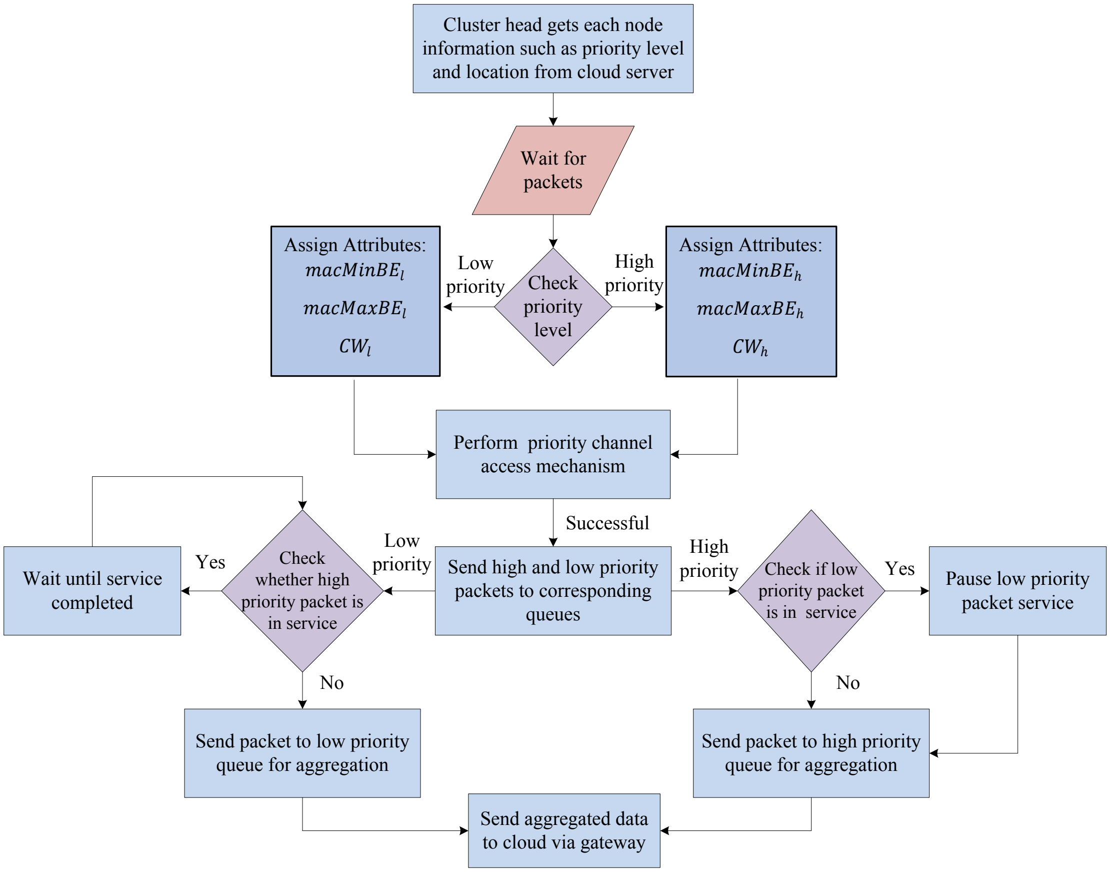
Cluster head gets each node information such as priority level and location from cloud server
Wait for packets
Assign Attributes: $macMinBE_l$ $macMaxBE_l$ CW_l
Low priority
Check priority level
High priority
Assign Attributes: $macMinBE_h$ $macMaxBE_h$ CW_h
Perform priority channel access mechanism
Successful
Wait until service completed
Yes
Check whether high priority packet is in service
Low priority
Send high and low priority packets to corresponding queues
High priority
Check if low priority packet is in service
Yes
Pause low priority packet service
No
No
Send packet to low priority queue for aggregation
Send packet to high priority queue for aggregation
Send aggregated data to cloud via gateway

Figure 2.4: Workflow diagram of the proposed scheme.

the non-linear equations (2.6), (2.7), and (2.8), we can obtain the corresponding values of α, P_{loss}, and $E[D]$.

2.3.3 Data Aggregation with Prioritization

In the case of the data aggregation with prioritization, the prioritized M/G/1 queuing model holds P priority class of data. The packets with the ith priority have arrival rate λ_i, $i \in \{1, 2, \ldots, P\}$, and follow the Poisson distribution. The lower value of i indicates a high priority packet type. In the system model, a preemptive priority rule is implemented, i.e., the new arrival of class ith priority packet will immediately preempt lower priority data currently being served and get the access to the services. The workflow diagram of the proposed scheme is presented in Fig. 2.4.

The waiting time W_i of the ith priority packet is the time spent in the queue before being served at the CH. The mean residual service time for the packets currently being served and the service time of the CH are denoted by R_i and S_i, respectively. The overall system delay is given by the summation of the waiting time and the service time of the packets. By using the Little's law, the expected waiting time of the ith priority packet is given by

$$E[W_i] = \frac{\sum_{j=1}^{i} \rho_j E[R_j]}{(1 - (\rho_1 + \ldots + \rho_i))(1 - (\rho_1 + \ldots + \rho_{i-1}))}, \quad i \in \{1, 2, \ldots, P\}, \tag{2.9}$$

where $\rho_i = \lambda_i E[S_i]$, $E[S_i]$ is the expected service time, and $E[R_i]$ represents the expected residual time. Let $E[\hat{S}_i]$ and $E[\hat{D}_i^{sys}]$ are the expected service time of ith priority packet by considering the interruptions of higher priority packet and the expected system delay in the ith priority queue respectively, and are calculated by

$$E[\hat{S}_i] = \frac{E[S_i]}{(1 - (\rho_1 + \ldots + \rho_{i-1}))}, \tag{2.10}$$

$$E[\hat{D}_i^{sys}] = E[\hat{S}_i] + E[W_i]. \tag{2.11}$$

In addition, the service time $E[S_i]$ of the CH, the expected residual time $E[R_i]$, and the second-order moment of the service time $E[S_i^2]$ for the priority-based data aggregation can be expressed as [30]

$$E[R_i] = \frac{2}{3} \lambda_i E[S_i]^2. \tag{2.12}$$

$$E[S_i^2] = \frac{4}{3} E[S_i]^2, \tag{2.13}$$

Similarly, the value of $E[S_i]$ can be calculated by using (2.5), (2.6), (2.7), and (2.8) in accordance with the values of CW_i and λ_i.

2.4 Performance Analysis

Table 2.1: Main Parameters Used in PHY layer

Parameters	Value
Noise Figure	23 dB
Bandwidth	30 kHz
Pathloss Exponent	4
Noise	15 dB
Preamble Length	40 bits
Transmission Power	5 dBm

Table 2.2: Main Paramenters Used in MAC Layer

Parameters	Value
Max Backoff Exponent	5
Min Backoff Exponent	3
Max CSMA Backoff	4
MAC Frame Payload	800 bits
Queue Size	51 frames
Data Rate	19.2 kbps
ACK Size	88 bits
MAC Overhead	48 bits
σ	0.32 ms
T_{ACK}	0.352 ms
T_{Tx}	1.12 ms
T_{CCA}	0.25 ms
T_{turn}	0.192 ms

In this section, we evaluate and analyze the performance of the proposed scheme in terms of the expected system delay and system reliability. The PHY layer and MAC layer simulation parameters are listed in Table 2.1 and Table 2.2, respectively [36]. The MATLAB software is used in order to obtain the results presented in this chapter.

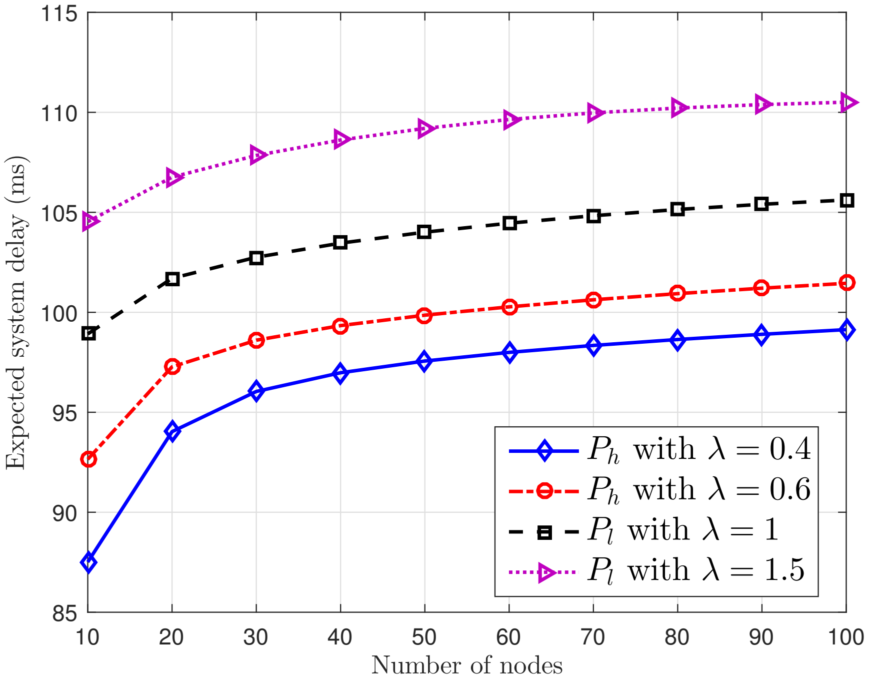

Figure 2.5: Performance evaluation of the proposed priority approach in terms of the expected system delay.

2.4.1 Expected System Delay

Figure 2.5 presents the expected system delay of packets with different priority levels versus the number of sensor nodes. The expected system delay of both high and low priority packets increases as the number of node increases because aggregation of higher number of data packets yields the longer service time. The low priority packets have the longer delay as compared to that of the high priority packets because the service time must accommodate the interruptions of all packets with the higher priority. Similarly, Fig. 2.6 shows the performance comparison of the proposed priority scheme with the non-priority scheme. The non-priority scheme has a similar characteristic curve; however, the delay is higher than the priority scheduling approach. Moreover, due to the prioritized channel access mechanism and preemptive priority rule, the high priority packets do not get any interruptions from the low priority packets and hence, the expected system delay is reduced.

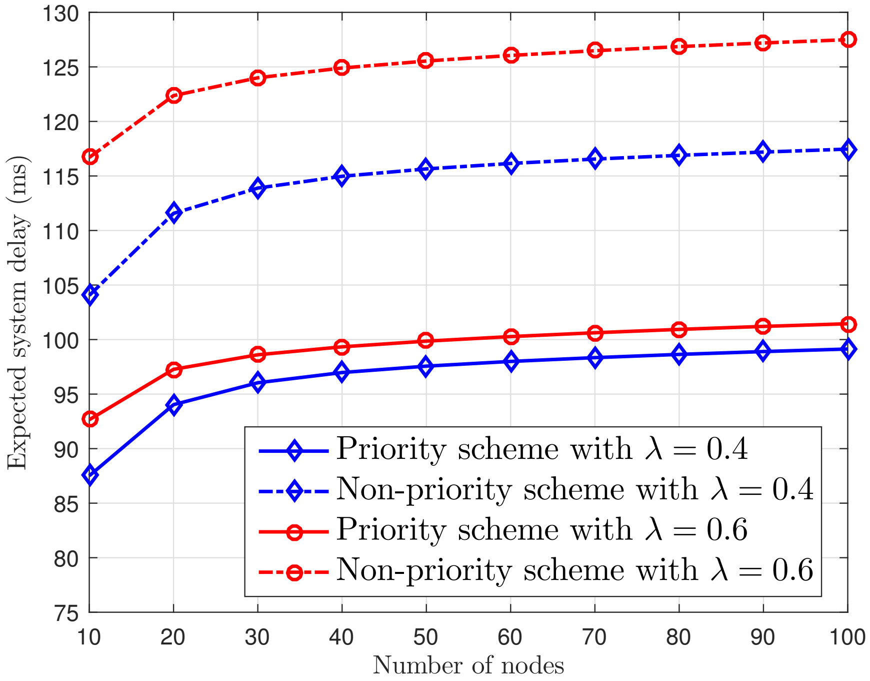

Figure 2.6: Performance comparison of the proposed priority approach with non-priority scheme in terms of the expected system delay.

2.4.2 Reliability

The proposed scheme is modeled as the preemptive M/G/1 priority queue with the system size K and each queue receives data frames by following the Poisson arrival process with the rate of λ data packets per second. The steady state probability that i data packets are present in the queue is given by [36]

$$p_i = \frac{\rho^i}{\sum_{j=0}^{K} \rho^j}. \tag{2.14}$$

The different possibilities that sensor nodes may not be able to successfully send data packets to the CH include: (i) if the buffer is full, (ii) if nodes fail to find the idle channel, and (iii) the packets are discarded after exceeding retry limits. By considering these aspects, the system reliability η can be calculated as [36]

$$\eta = (1 - p_k)(1 - P_{cf})(1 - P_{cr}), \tag{2.15}$$

where p_k is the probability of having full buffer with k frames and is given by (2.14), P_{cf} is the probability that the packet is dropped due to the channel access failure, and P_{cr} is the probability of packet discarded due to the retry limit.

Figure 2.7 depicts the overall system reliability versus the number of nodes. It is clearly observed that the network reliability decreases as the number of nodes increases. Each node in the queue begins to experience the congestion problems due to a large number of nodes; collisions become more frequent, and the packet re-transmissions are more recurrent. Subsequently, the delays get longer as the queues become busier. The probability of frame loss also increases because of the collisions, the retry limits, and the link constraints. Moreover, due to the employed priority-based channel scheduling mechanism and queuing policy, the network reliability of the high priority nodes is noted to be higher than that of the low priority nodes.

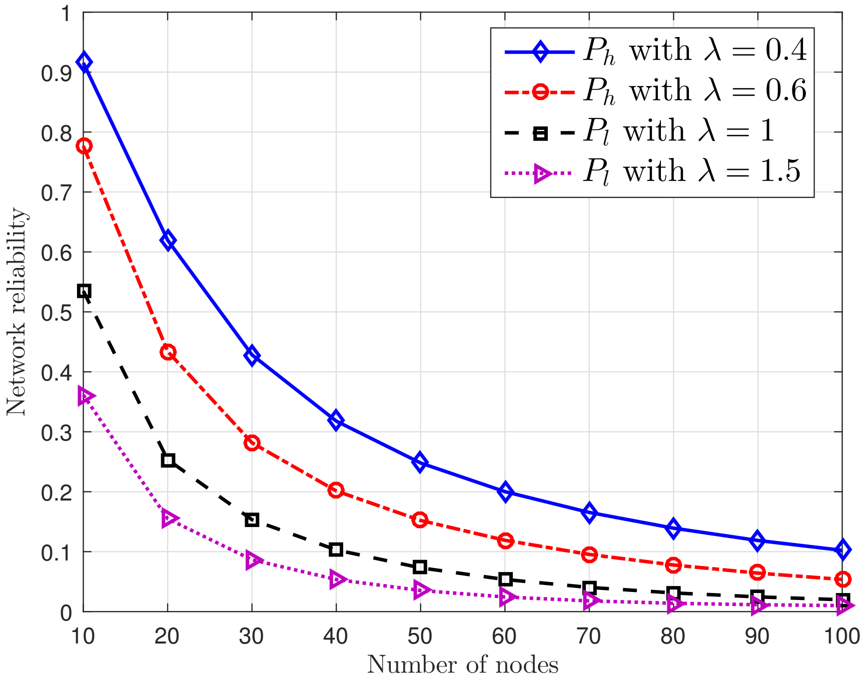

Figure 2.7: Performance evaluation of the proposed priority approach in terms of network reliability.

2.5 Chapter Summary

IoT networks consist of a large number of sensor nodes for different sensing and monitoring purposes. The resource-constrained IoT devices operating in highly dense networks may be affected by the data collisions, packet loss, packet delays and low network throughput. These IoT devices usually have diverse data traffic with different latency and system reliability requirements. In this chapter, a cloud-assisted latency minimization scheme is proposed by using prioritized channel access and data aggregation at the CH. In addition, the joint impact of packet scheduling and aggregation is considered by using the preemptive M/G/1 queuing model. With the help of numerical results, it has been shown that the prioritized channel access and data aggregation scheme provides substantial improvements in terms of latency and system reliability as compared to the non-prioritized scheme. In the future work, we plan to use network simulator tools to analyze the performance of the proposed scheme in real-world IoT applications such as eHealthcare and industrial automation.

Chapter 3

Device Grouping for Efficient Channel Access in IoT Networks

3.1 Introduction

The emergence of IoT has changed the perspective of wireless communications since the number of devices has exponentially increased over the recent years. In order to support these massive number of devices in the existing networks, the IEEE Task Group ah (TGah) is dedicated to the standardization of a new IEEE 802.11ah protocol, which is customized for the large-scale networks [37]. The IEEE 802.11ah is an emerging wireless standard in sub-1 GHz license exempt bands for cost effective and range-extended communication. This standard adopts the grouping-based Medium Access Control (MAC) protocol to reduce the contention overhead [38]. Moreover, by utilizing the Multiple Input Multiple Output (MIMO) scheme, the IEEE 802.11ah based systems can utilize the benefits of spatial diversity to enhance the link capacity and to extend the coverage area [37].

The recent advances in IoT have led to numerous emerging applications ranging from eHealthcare to industrial control, which often demands stringent QoS requirements such as low-latency and high system reliability. However, the ever-increasing number of connected devices in ultra-dense IoT networks and the dynamic traffic pattern increase the channel access delay and packet collision rate. In this regard, we propose a sector-based device grouping scheme for fast and efficient channel access in IEEE 802.11ah based IoT networks such that the total number of the connected devices within each sector is dramatically reduced. We propose to employ a sectorized grouping scheme in IEEE 802.11ah based ultra-dense IoT networks by employing a spatial orthogonal access scheme. The main objectives of the proposed sectorized grouping scheme are to reduce the channel contention by reducing the number of stations within a sector, to enable the spatial sharing of Random Access Window (RAW) slots among the overlapping APs/other neighbor stations in different groups, and to minimize the hidden station problem [39]. In the proposed scheme, the cloud-center provides the stations information to the AP via an Internet link since it has a global knowledge of the network. The AP then broadcasts beacons to different geographical locations by utilizing simple sectorized beams. The number of stations is further divided into different groups uniformly within the sectors [38]. Thus divided groups are assigned to different RAW slots for the data transmission and they spatially access the channel for data transmission towards the AP. By considering this set-up, the performance of the proposed sectorized grouping scheme is analyzed in terms of system delay and network throughput. Finally, the performance of the proposed scheme is compared with conventional DCF and 802.11ah without sectorization via numerical results.

The rest of this chapter is organized as follows. Section 3.2 summarizes the related works. In Section 3.3, a brief description of IEEE 802.11ah and its grouping mechanism are presented. In Section 3.4, the overall system model and the proposed sector-based grouping scheme are described. In Section 3.5, the probability of transmission, throughput, and delay analysis of the proposed method are presented. In Section 3.6, the performance of proposed scheme is evaluated via simulations. Finally, the chapter is concluded in Section 3.7.

3.2 Related Works

Although the grouping based MAC protocol in IEEE 802.11ah standard significantly reduces the number of collisions and the contention overhead, existing solutions consider the grouping of devices/stations by using simplistic approaches either in a random or an uniform manner. In [40,41], the authors proposed an optimal group division and resource allocation strategy for static network conditions. For the group based contention mechanism, the size of a group is the key design parameter since the number of stations significantly affects the network performance. However, the IEEE 802.11ah standard does not provide any guidelines for the group size. In this regard, the authors in [42] provided an expression to find the optimum group size on the basis of the number of active stations per group, traffic arrival rate, and the beacon interval. Nevertheless, in most of the existing solutions, devices are randomly assigned to different groups, and less attention has been given to the formation of efficient and reliable groups.

3.3 Grouping Scheme in IEEE 802.11ah Standard

The IEEE 802.11ah standard is designed to support the applications with a large number of communicating devices, extended coverage area, and low energy consumption [43]. This new standard maintains similar network architecture as IEEE 802.11 for fixed, outdoor and point-to-multi-point applications. In order to meet the above mentioned requirements, IEEE 802.11ah differs from the traditional IEEE 802.11 in both the physical (PHY) and MAC layers. The PHY layer of IEEE 802.11ah is based on the IEEE 802.11ac and is a tenfold down-clocked version of IEEE 802.11ac, operating over a set of unlicensed radio bands (all sub-1 GHz) [44]. A single-user beamforming, MIMO, and downlink multi-user MIMO first presented in IEEE 802.11ac are also adopted in the IEEE 802.11ah standard [37].

In order to reduce the contention level in M2M networks with the thousands of devices, IEEE 802.11ah has introduced a new grouping based MAC protocol. This protocol also supports the advanced power saving mechanism, improved medium access, and throughput enhancement [15]. In legacy IEEE 802.11 networks, the AP can support only up to 2008 stations. In addition, the Traffic Indication Map (TIM) bitmap imposes another limitation. However, the TGah has extended the range of Association ID (AID) numbers from 0-8191, and the length of TIM is also increased to 8192 bits to support a large number of stations [45]. In particular,

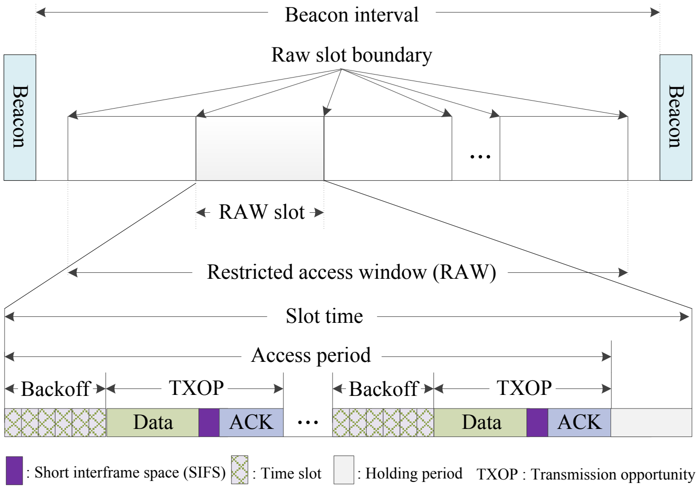

Figure 3.1: Grouping-based MAC protocol in IEEE 802.11ah standard.

sensors in a wireless network are partitioned into several groups. The channel access time is partitioned into beacon intervals, each of which is further divided into a number of equal duration RAW slots as depicted in Fig. 3.1. Each RAW slot is then assigned to a group of sensors, and only the devices within a particular group are allowed to access the RAW slot assigned to that group. Since only a part of stations contend for the channel access in a particular RAW slot, the collision probability becomes significantly lower as compared to the conventional IEEE 802.11ah.

3.4 System Model and Proposed Grouping Mechanism

The future of networking is not only bounded with people but also is related to the integration of all objects, media and services, thus creating the Internet of Everything (IoE) [46]. The IoE connects the people, communicating objects or things, processes, and data in an effective manner to provide the ubiquitous services [2]. In this regard, we consider a large-scale IoT scenario composed of N number of stations deployed over a circular area of radius 1 km as depicted in Fig. 3.2. The IoT network is organized as a hierarchical model, in which the AP is responsible to collect the information sensed by different stations/sensors. The data

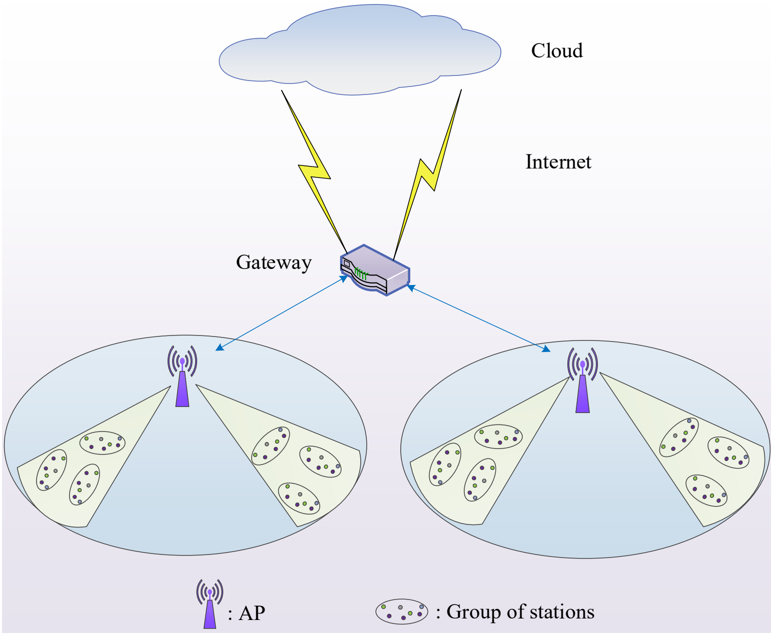

Figure 3.2: Pictorial representation of the proposed grouping mechanism.

packets from each station are delivered to the AP in one hop. The gateway is then responsible for transferring the collected information to the core network or cloud via an Internet link. Similarly, the cloud-center provides the location information of each station to the AP. The gateway and AP are considered to be positioned at the specific geographic locations [47]. The pictorial representation of the proposed sector-based grouping scheme is shown in Fig. 3.2. In this scheme, the AP broadcasts beacons to the specific locations by utilizing the sectorized beams created with multiple antennas at the AP. The stations thus can take the advantages of distinct geographical areas since the AP coverage area is divided into different sectors.

In the considered scenario, the channel access time is divided into different mini-slots. We consider K groups of stations, each with the size of g_k and $\sum_{k=1}^{K} g_k = N$, $k = 1, 2, 3, ..., K$. Figure 3.2 shows the geographically sectorized areas of the APs using multiple beamforming antenna arrays. The data transmission from the stations in one sector to the AP does not interfere with the transmission in other sectors. Let $\varkappa$ be the number of stations in the kth group of a particular sector. The sectorized beam of the AP covers different groups of size g_k. For the given station $s_{i,k}$, $i = 1, 2, 3, ..., \varkappa$ in group k of the given sector within a single AP, we can write the following relation as in [39]

$$\sum_{i=1}^{\varkappa} s_{i,k} \ll \sum_{\forall k \in AP} g_k. \tag{3.1}$$

The frame durations of a RAW and a RAW slot are denoted as Γ_r and Γ_s, respectively. Similarly, $\Gamma_{s,k}$ denotes the time duration of the RAW slot allocated to the kth group, and therefore, $\Gamma_r = \sum_{k=1}^{K} \Gamma_{s,k}$. The stations in a specific group periodically access the channel in the specified RAW slots. The proposed group formation procedure is detailed later in Algorithm 1 (Section 3.5).

3.5 Probability of Transmission, Throughput and Delay Analysis

In this section, the probability of transmission, throughput and delay analysis of IEEE 802.11ah RAW for both the conventional Distributed Coordination Function (DCF) and the proposed approach is presented. The DCF defines two medium access mechanisms to transmit the data packets, i.e., the basic access and Request-To-Send (RTS)/Clear-To-Send (CTS). A station with the data packet senses the channel activity before transmitting towards the AP. If the channel is idle, a station transmits a packet immediately. However, if the channel is busy, the station persists to monitor the channel until it becomes idle for more than DCF Interframe Space (DIFS). To minimize the collision probability, the station waits until the random backoff interval before transmitting a packet. In the case of RTS/CTS, the short RTS and CTS packets are exchanged to reserve the channel prior to the data packet transmission. The neighboring stations will refrain the data packet transmission until the ongoing transmission is complete. More specifically, the RTS/CTS mechanism minimizes the data collision probability, collision duration, and also copes with hidden stations. The RTS/CTS scheme enhances the MAC protocol performance by minimizing the collision probability, and thus is more advantageous for ultra-dense IoT networks [48].

Let n be the number of contending stations for the medium access in the given RAW slot. In the case of saturation conditions, each station immediately transfers the packet after the completion of each successful transmission. However, due to the consecutive transmissions, each packet needs to wait for the random backoff interval before transmitting. At each transmission attempt, each packet collides with a constant and independent probability p. Moreover, when the number of stations tries to access the slot, devices are contended by the random back off procedure. Similarly, m and CW_{min} be the maximum back-off stage and minimum value of contention window, respectively. Then the maximum value of contention window becomes $CW_{max} = 2^m CW_{min}$ [49]. The transmission occurs only when the back-off counter becomes zero. The probability that a station transmits a packet in a randomly chosen slot is expressed as [49]

$$P_\tau = \frac{2(1-2p)}{(1-2p)(CW_{min}+1) + pCW_{min}(1-(2p)^m)}. \tag{3.2}$$

Similarly, the conditional collision probability p when a station transmits a packet is given by

$$p = 1 - (1 - P_\tau)^{(n-1)}. \tag{3.3}$$

Subsequently, the probability p_{tr} that there is at least one ongoing transmission in the considered time slot is calculated as

$$P_{tr} = 1 - (1 - P_\tau)^n. \tag{3.4}$$

Next, the probability P_s that a packet transmission occurring on a channel becomes successful is expressed as

$$P_s = \frac{nP_\tau(1-P_\tau)^{(n-1)}}{1-(1-P_\tau)^n}. \tag{3.5}$$

The normalized system throughput is defined as the fraction of time that a random access channel is used to successfully transmit the payload bits [50], and can be expressed as

$$S_c = \frac{P_s P_{tr} E[P_{ld}]}{(1-P_{tr})\sigma + P_{tr}P_sT_s + P_{tr}(1-P_s)T_c}, \tag{3.6}$$

where $E[P_{ld}]$ is the average payload size, T_s is the average time slots of successful transmission,

T_c is the average time slots of an unsuccessful transmission, and σ is the backoff slot duration, respectively. The values of T_s and T_c can be calculated as [50]

$$T_s = \begin{array}{l} T_{RTS} + T_{CTS} + T_{PHY_{hdr}} + T_{MAC_{hdr}} + E[P_{ld}]+ \\ T_{ACK} + 3T_{SIFS} + T_{DIFS} + 4\delta, \end{array} \tag{3.7}$$

$$T_c = T_{RTS} + T_{DIFS} + \delta, \tag{3.8}$$

where T_{RTS}, T_{CTS}, $T_{PHY_{hdr}}$, $T_{MAC_{hdr}}$, and T_{ACK} are the transmission times of RTS, CTS, PHY header, MAC header, and ACK frame, respectively. The T_{SIFS} and T_{DIFS} define the short and distributed inter-frame space durations, and δ is the propagation delay.

The above analysis is for the conventional DCF. Based on conventional DCF, the authors in [15] proposed centralized uniform and decentralized random grouping schemes without considering the geographical locations of stations. The conventional DCF is a distributed channel access mechanism for the standard Wireless Local Area Network (WLAN) to provide the long-term fair channel access for the stations. However, dense IoT stations may experience unfair channel allocations on the basis of their locations because of the propagation characteristic of radio signals, such as capture effect and path attenuation. The stations situated nearby an AP can access the channel several times more than a station located far from the AP. The authors in [49] assume that the collision probability p is identical for all the stations. Consequently, the probability P_τ that a station transmits a packet in a given idle slot is also same for the all stations. However, in the realistic scenario, the probability of collision p also depends on the distance from the AP. The stations far from the AP experiences high collisions as compared to the stations located nearby the AP. Thus a station located nearby an AP favorably transmits more data packets as compared to the station that is far from the AP. To address this aspect, our analysis incorporates the location of stations and the effect of the AP coverage sectorization.

The proposed grouping mechanism and channel access mechanism is presented in Algorithm 1. In the proposed sectorized grouping scheme, the stations in the same sector can hear each other's transmission, and thus the hidden station problem is minimized [43]. Moreover, due to the sectorized beamforming, a large number of packets can be transmitted successfully towards the AP due to the lower packet collision rate. Hence, the throughput presented in (3.6) for the sectorized scenario can be expressed as [39]

$$S_{c,sector} = \frac{P_s P_{tr} E[P_{ld}]}{(1 - P_{tr})\sigma + P_{tr} P_s T_s + P_{tr}(1 - P_s)\zeta}, \tag{3.9}$$

where ζ is the residual frame collision coefficient. The AP coverage sectorization reduces the frame collision probability, and the coefficient ζ can be expressed as

$$\zeta = p_{c_k} T_c. \tag{3.10}$$

The frame collision probability p_{c_k} can be derived as [39]

$$p_{c_k} = (1 - (1 - p_\eta)^{\vartheta(n-1)}), \tag{3.11}$$

with

$$p_\eta = 1 - \left(1 - \frac{1}{CW_{avg}}\right)^{n-1}, \tag{3.12}$$

where ϑ is the slot time and CW_{avg} is the average backoff window size.

The IEEE 802.11ah adopts the RAW scheme in order to cope with the packet collision, channel contention and the hidden station problem. A station is expected to contend for channel access during the allocated RAW slot. However, the stations belonging to the specified RAW slot may not hear each other's transmission in the ultra-dense IoT network, thus packet collision rate increases significantly. The sectorization involves the partitioning of the AP coverage area into different non-overlapping sectors by using antenna arrays. The frame collision coefficient due to the sectorized beam forming is given by (3.10), and the corresponding throughput can be calculated by substituting T_c with ζ in (3.6) and is expressed in (3.9). As presented in Algorithm 1, the AP further divides the stations of the each sector into different groups uniformly. As a consequence, the stations belonging to a specified group within a sector can only gain the channel access during the allocated RAW slot, thus minimizing the frame

Algorithm 1 Proposed group formation mechanism

```
Network is setup as shown in Fig. 3.2.
Cloud-center provides the device location information to the AP via the Internet link.
By analyzing the device location information, AP sectorizes its coverage area into different sectors and assigns a separate beam to each sector by using multiple antennas.
AP calculates the number of stations located in a particular sector as $\chi$, $n \leq \chi \leq N$.
AP further divides the stations within each sector into different groups uniformly as
$g_{k,j} \in \{g_{1,j}, g_{2,j}, ..., g_{K,j}\}$, $\sum_{k=1}^{K} g_{k,j} = \chi_j$ ; $g_{k,j}$ is kth group in jth sector.
Station authentication and association are performed by the AP using the procedure in [45].
AP assigns distinct RAW slots to different groups within each sector, and the devices within distinct groups in different sectors access the allocated RAW slots by employing a spatial orthogonal acces mechanism.
AP broadcasts network information including sector ID, group ID, and RAW information by utilizing the sectorized beams.
Stations within each group belonging to a particular sector spatially contend for the channel access in the assigned RAW slot by following IEEE 802.11ah standard.
```

collision probability given by (3.3) as compared to the conventional standard.

While analyzing the delay aspects, the average delay $E[D]$ for successfully transmitted data packet is given by [48]

$$E[D] = E[X]E[slot], \tag{3.13}$$

where $E[X]$ is the average number of time slot, and $E[slot]$ is the average length of a slot time for the successful data transmission. The term $E[X]$ can be calculated as [48]

$$E[X] = \frac{(1-2p)(CW_{min}+1) + pCW_{min}(1-(2p)^m)}{2(1-2p)(1-p)}. \tag{3.14}$$

Table 3.1: Network Paramenters

Parameters	Value
Payload	256 bytes
PHY Header	128 bits
MAC Header	272 bits
RTS Frame	PHY Header + 160 bits
CTS Frame	PHY Header + 112 bits
ACK Frame	PHY Header +112 bits
Time Slot	50 μs
SIFS	28 μs
DIFS	128 μs
CW_{min}	32
p_{c_k}	0.01
Propagation Delay	1 μs
Number of Sectors	4

3.6 Performance Analysis

In this section, the performance of the proposed sectorized and grouping scheme is analyzed and evaluated in terms of the throughput and system delay by using MATLAB. The simulation

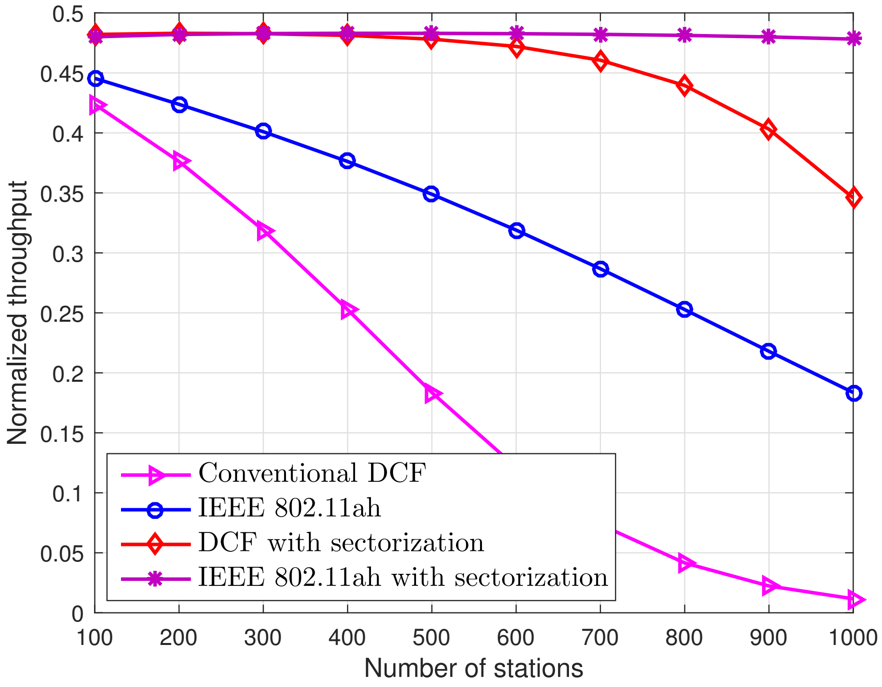

Figure 3.3: Performance comparison of the proposed sectorized grouping scheme with conventional DCF and IEEE 802.11ah mechanism in terms of the normalized throughput.

parameters are presented in Table 3.1.

3.6.1 Throughput

Figure 3.3 presents the normalized throughput of the proposed and conventional schemes versus the number of stations up to 1000. Similarly, Fig. 3.4 shows the performance comparison of the proposed priority scheme with the conventional DCF and IEEE 802.11ah for the ultra-dense IoT scenario, i.e., the number of stations up to 4000. In both the cases, the normalized throughput decreases as the number of stations increases because of the channel congestion and packet collisions. The DCF scheme is heavily influenced by the increasing number of hidden stations and packet collisions, which reduces the network throughput significantly. However, the throughput of IEEE 802.11ah is improved by reducing the number of stations in each group. Furthermore, the grouping based protocol decreases both the collision rate and the hidden ter-

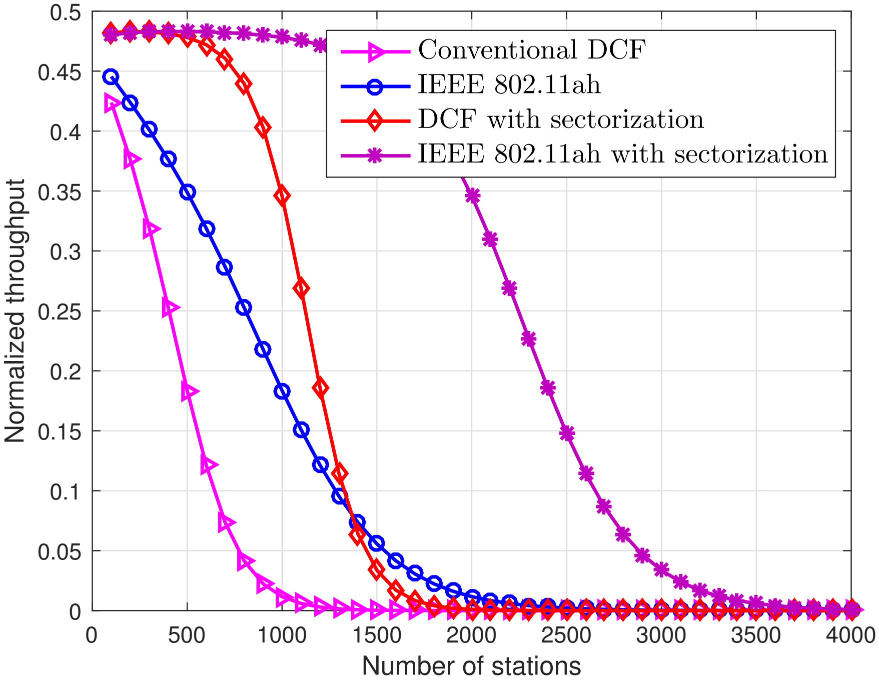

Figure 3.4: Performance comparison of the proposed sectorized grouping scheme with conventional DCF and IEEE 802.11ah mechanism in terms of the throughput for ultra-dense IoT scenario (i.e., number of stations up to 4000).

minals significantly. In addition, the sectorization of the AP coverage area further reduces the hidden stations and packet collisions by utilizing the distinct areas and simple sectorized beamforming. As a result, the proposed sector-based grouping scheme provides a significant throughput improvement by employing a spatially orthogonal access scheme.

Figure 3.5 shows the effect of the number of RAW slots (N_{RAW}) in throughput performance of the proposed sectorized grouping scheme and IEEE 802.11ah. It has been noted that the throughput of the overall network is enhanced while increasing the value of N_{RAW} from 2 to 5. The number of RAW slots allows to limit the number of contending stations in a given interval of the time. However, relatively higher value of N_{RAW} in IEEE 802.11ah is not enough to deliver the packets successfully because of the shorter time slot. In the proposed sectorized grouping scheme, stations within different groups can access the distinct orthogonal wireless channels with the reduced collision probability.

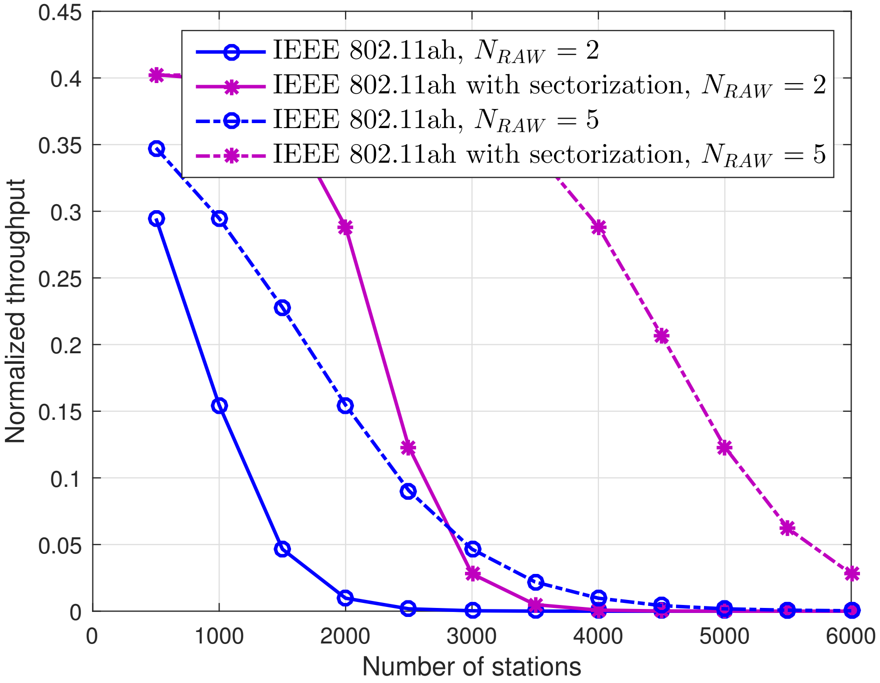

Figure 3.5: The effect of number of RAW slots in throughput of the IEEE 802.11ah with and without sectorized grouping scheme.

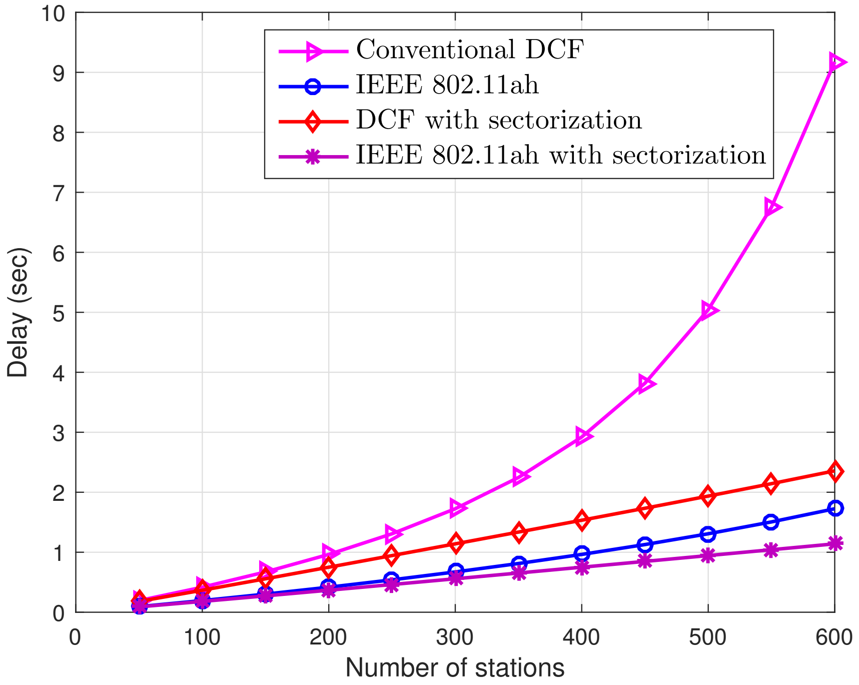

Figure 3.6: Performance comparison of the proposed sectorized grouping scheme with conventional DCF and IEEE 802.11ah mechanism in terms of the system delay.

3.6.2 System Delay

Figure 3.6 depicts the overall system delay of the network versus the number of stations. It is clearly observed that the system delay increases as the number of stations increases. The intuitive reason behind this trend is the following. Each station in the network begins to experience the congestion problems because of a large number of stations. The number of packet collisions becomes more frequent, and the data re-transmissions are more recurrent. The probability of packet loss also increases due to the link constraints, the retry limits, and collisions.

Also, from Fig. 3.6, it can be depicted that the conventional DCF scheme has the higher delay as compared to the IEEE 802.11ah and the proposed scheme. In addition, the proposed sectorized grouping mechanism enhances the delay performance of IEEE 802.11ah. This performance gain is achieved due to the reason that the proposed scheme reduces the frame collisions by utilizing the spatially orthogonal access scheme, and by minimizing the number of hidden terminals.

3.7 Chapter Summary

Future IoT networks are expected to support a massive number of stations/sensors in diverse applications with different QoS requirements. However, the heterogeneous IoT devices operating in ultra-dense network scenarios may be affected by the packet collisions, delays, and low network throughput. In this chapter, a sector-based device grouping scheme is proposed for IEEE 802.11ah based IoT networks. In the presented framework, the cloud-center facilitates the grouping process by providing the stations' location information to the AP. The AP forms the sectors, and divides into different groups according to the number of stations and their corresponding locations. In addition, the sector-based grouping allows the substantial improvement on packet collision rate/probability and the throughput of ultra-dense IoT networks by utilizing the spatially orthogonal access mechanism. Via numerical analysis, it has been shown that the proposed sector-based grouping mechanism significantly improves the network throughput and system delay as compared to the conventional DCF and IEEE 802.11ah grouping scheme.

Chapter 4

Time Synchronization protocol for Event Critical Applications in Wireless IoT

4.1 Introduction

The emerging Internet of Things (IoT) paradigm is expected to interconnect various objects and processes for massive information collection, analysis, and utilization [5]. The total number of connected sensors and machine type communication nodes has been rapidly increasing over recent years, and leading to dense wireless IoT networks. The interconnectivity, mutual interference, security, and synchronization are the open issues in dense wireless IoT networks. Precise clock synchronization is one of the crucial issues in a wide range of distributed wireless IoT applications to perform event driven sensing and control. In distributed IoT networks, high synchronization precision is needed to maintain the common reference time for collaborative information exchange and data fusion [51,52]. The density and closeness of sensor nodes make wireless IoT network more susceptible to packet congestion. The probability of packet loss also increases because of the collisions, the link constraints, and the retry limit [53]. Subsequently, the uncertainties arising in packet delays affect the overall clock synchronization process. To overcome the uncertainties that arise in the packet delivery process, a series of standards such as IEEE 802.15.4e [54], IEEE 802.11ah [55] and WirelessHart [56] have been proposed in recent years. Regardless of their extensive applications in industries, there is a lack of effective mechanisms in supporting the real time industrial applications. As presented in ISA 100, the data traffics in the automation and control applications are classified into safety, control, and monitoring based on the reliability and latency requirements [54]. The safety data traffic refers to emergency situations such as fire alarm and automatic shutdown. This kind of data traffic is more critical in nature and requires reliable synchronization to the controller within stringent deadlines. Failing to meet the synchronization bound for this kind of traffic may cause system instability and also pose a threat to human safety.

The major source of synchronization error in industrial IoT networks present in packet delivery is caused by the channel access process. The non-deterministic delays in packet delivery caused by the channel access time can be much larger than the required synchronization precision for the event critical applications [57]. The Medium Access Control (MAC) layer is primarily responsible for channel access of nodes within a network that uses a shared medium. In the Time Division Multiple Access (TDMA), the node must wait for the specified time slot before transmitting the packet, whereas in a contention-based mechanism such as Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), nodes must wait until the channel is clear before transmitting. In addition, the Request-To-Send (RTS)/ Clear-To-Send (CTS) mechanism is also required to exchange the packets.

The non-deterministic random back off delay caused by the channel access mechanism affects the accuracy and precision of the clock synchronization. The IEEE 802.15.4 is a widely used standard for wireless IoT networks, which adopts a hybrid protocol with CSMA/CA and TDMA for channel access [58]. In the Contention Access Period (CAP), CSMA/CA is used for the channel access, whereas in the Contention Free Period (CFP), TDMA scheme is used for data transfer by providing a number of Guaranteed Time Slots (GTS). The GTSs can provide the deterministic channel access for the time critical nodes, however it has following disadvantages: (i) the number of GTSs are limited to seven (i.e., GTS starvation), (ii) to use the GTS, the node has to send GTS request packet in CAP, and wait till the allocation confirmation in the next beacon period, and (iii) the GTS allocation based on the First Come First Served

(FCFS) scheme does not provide a prioritized channel access for critical traffic, and also is not suitable for the event critical applications [59]. The range of access delays varies from 10 to 500 ms [60], which is significantly larger than other delay factors. To compensate the non-deterministic packet latency, the synchronization protocol designed for wireless IoT should fulfill the limited resources requirements of IoT applications.

The critical synchronization challenge at MAC layer for dense IoT communication lies in facilitating the guaranteed channel access to the extremely large number of nodes by supporting the unique traffic characteristics and diverse service requirements. In a dense IoT network, the series of timestamp messages transmission is required to estimate the relative clock offsets and skews between IoT nodes. In general, the time synchronization in an IoT network can be regarded as the process of estimation and mitigation of the random latencies during the message transmission in a wireless channel. In wireless IoT networks, several synchronization mechanisms use the MAC layer timestamping procedure to reduce the latency uncertainty. The timestamping at the MAC layer can be done just before the transmission of the packet or immediately after the packet has been received. As a consequence, the non-deterministic delay can be estimated.

In this Chapter, an efficient clock synchronization scheme is proposed for the irregularly deployed sensor nodes to maximize the synchronization precision of the event critical applications, and to further enhance the reliability of overall IoT networks. All the sensor nodes associated with the coordinator get the reference clock information during the network initialization and perform clock synchronization in a beacon interval. The critical nodes also perform the synchronization with every event detection to enhance the clock precision. In the proposed scheme, a priority-based deterministic channel access mechanism is employed to reduce the access latency by assigning different MAC layer attributes. Also, an Emergency Indication (EI) slot is used to prioritize the critical traffic over normal traffic. Finally, the performance of the proposed scheme is analyzed, and the synchronization precision of the event critical nodes is compared with normal nodes.

The rest of this chapter is organized as follows. Section 4.2 presents the sources of clock synchronization error in detail. Section 4.3 summarizes the related works and provides the literature review in the area of synchronization. Section 4.4 describes the system model and proposed scheme in detail. Section 4.5 evaluates the performance of the proposed method via simulations. Section 4.6 concludes this chapter.

4.2 Sources of Clock Synchronization Error

In wireless IoT networks, the uncertainties in the packet delays cause clock estimation errors [61]. The delay sources in the packet transmissions that affect the synchronization process are as follows;

i. Send Time, T_s: The overall time required to prepare the packet at the application layer and send to the network layer, which may include the delay introduced by the operating system. The nature of this delay is non-deterministic.

ii. Access Time, T_a: This delay is introduced at MAC layer during the channel access process. This is one of the major components in synchronization and also highly variable depending upon the application specific protocol.

iii. Transmission, T_t/ Reception Time, T_{rp}: The time required for transmitting/ receiving a message at the physical (PHY) layer. This delay depends on the packet size and the rate of the wireless channel, and is deterministic in nature. The illustration of the packet delivery latency

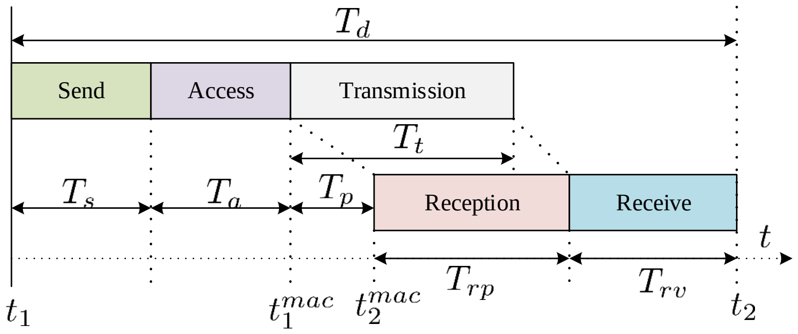

Figure 4.1: Illustration of packet transmission and reception delays between two IoT nodes.

Table 4.1: Delay sources in packet transmissions

Delay factor	**Value**	**Nature**
Send/Receive	0-100 ms	Non-deterministic, Depends on processor capacity
Access	10-500 ms	Non-deterministic, Depends on channel contention
Transmission/Reception	10-20 ms	Deterministic, Depends on packet size
Propagation	<1μs (up to 300 meters)	Deterministic, Depends on distance
Interrupt Handling	<5μs (in most cases)	Non-deterministic, Depends on interrupts
Encoding/Decoding	100-200μs	Deterministic, Depends on radio chipset
Byte Alignment	0-400μs	Deterministic, Can be calculated

over the wireless link is presented in Fig. 4.1.

iv. Propagation Time, T_p: The actual time required to transmit a message from sender to receiver through the wireless channel.

v. Receive Time, T_{rv}: The overall time spent to encode and transmit a received message to the application layer at the receiver side.

The total delivery delay, T_d can be calculated as

$$T_d = T_s + T_a + T_p + T_{rp} + T_{rv}. \tag{4.1}$$

The above delivery delay is directly related to the synchronization error. The delay can be compensated for as long as it can be computed accurately. The latency components can be classified into two categories; deterministic (i.e., fixed components) and non-deterministic/stochastic (i.e., random or variable components) [62]. The variable components depend on several network factors such as network traffic, channel conditions, and the number of nodes, and thus, there is no single mathematical model to estimate errors that fit every application scenario. The *Propagation time* is negligible in most of the IoT scenarios, as it is much smaller than the clock resolution. The *Reception Time* and *Transmission Time* are dependent on the message length and radio frequency. However, *Send time*, *Receive time*, and *Access time* are rather un-

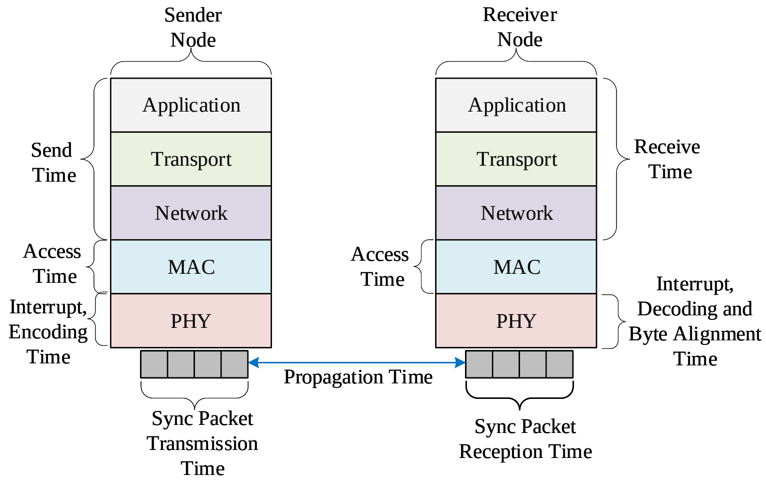

Figure 4.2: Sources of delay uncertainties in packet delivery process.

predictable, causing the delay uncertainty on the IoT networks. The values and the nature of the various latencies are summarized in Table 4.1 [60]. The delay uncertainties in the packet delivery process are depicted in Fig. 4.2.

4.3 Related Works

In a distributed IoT network, each node has its own internal clock, and operates in different oscillation frequencies. In real time, the clock may drift in a micro-second scale with other nodes, and clock error can be accumulated over the time. As a consequence, IoT nodes may not operate in a synchronized state. In the traditional distributed wireless system, to overcome the clock synchronization issues, two basic schemes are adopted in the literature. The first scheme deals with the physical clock of the nodes, and the second scheme deals with the logical clock. In physical clock synchronization, each node agrees to tune to a common clock value, whereas, in logical clock synchronization, the knowledge of chronological ordering of real events is required [63]. Network Time Protocol (NTP) is widely used on the Internet for physical clock synchronization [64]. The NTP synchronizes the nodes to the Coordinated

Universal Time (UTC), and follows the hierarchical client-server architecture. However, this protocol is designed for the traditional wired system, and does not suit for dense IoT due to the limited battery life and dynamic topology. The remote clock reading scheme is proposed to estimate the non-deterministic message latency by considering the client-server architecture [65]. The main drawbacks of this scheme are: (i) message complexity, and (ii) time uncertainty due to the network routing and traffic conditions.

In [66], the authors proposed Time-sync Protocol for Wireless Sensor Network (TPSN) based on two-way message exchange between the sender and receiver. The pairwise synchronization is performed at the edge of the hierarchical model. However, the TPSN suffers from non-deterministic message delay from the sender, and also is unable to perform the relative skew estimations and corrections. The Flood Time Synchronization Protocol (FTSP) is proposed for WSNs based on the one-way broadcast message from a sender to the multiple receivers [60]. At first, the global clock reference node is elected, and a spanning tree is built at the reference node. Afterwards, each node synchronizes with its parent node. The major limitations of FTSP are: (i) suffer from substantial overhead to elect the new root node if the root dies, and (ii) non-deterministic synchronization error due to the message route in the constructed tree. Based on the FTSP, authors in [67] presented the Ratio-based time Synchronization Protocol (RSP) to improve the estimation accuracy by eliminating the message delay. The RSP uses the two synchronization packets to estimate the clock drift of the receiver with the sender (i.e., root node).

A Reference Broadcast Synchronization (RBS) based on the receiver to receiver synchronization for WSNs is presented in [68]. The nodes calculate the relative offset and skew based on the reception times of the beacon message. Nevertheless, this scheme uses a series of message exchange to estimate both relative skew and offsets. The RBS is not appropriate for self-organized IoT networks due to the temporal data path failure and high probability of packet collisions. The Pair-wise Broadcast Synchronization (PBS) based on the Receiver Only Synchronization (ROS) and Sender-Receiver Synchronization (SRS) is proposed in [69] to achieve network-wide synchronization. In [70], the authors presented the Relative Reference less Receiver-Receiver Synchronization (R^4Sync) based on the receiver-to-receiver synchronization scheme. To eliminate the single point failure problem in the RBS, this scheme allocates the reference clock function to all nodes instead of only one node. The authors in [14] introduced an extended and improved Emergent Broadcast Slot (EBS) scheme to guarantee robust communication between nodes. The EBS mechanism is fully decentralized, which facilitates collaboration between nodes within the duty-cycle to avoid packet collisions. Nevertheless, in most of the existing schemes, channels and time slots are randomly assigned for the packet transmissions, and less attention has been provided to the guaranteed synchronized channel access for the event critical sensor nodes.

4.4 System Model and Proposed Schemes

In this chapter, an industrial wireless IoT scenario with N number of sensor nodes is considered to implement the proposed scheme. A high level of synchronization accuracy is required

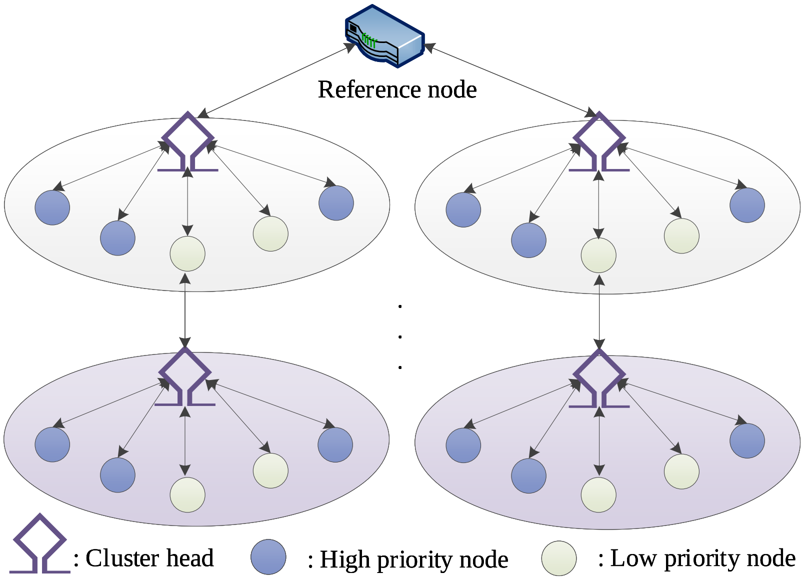

Figure 4.3: Pictorial representation of the network scenario. The sensor nodes, including Cluster Head (CH) and the intermediate nodes are organized in a hierarchical model.

to maintain the time consistency of critical sensing information, and to ensure the system reliability to fulfill the censorious requirements of the industrial systems. The data gathered by N nodes are divided into two classes, i.e., time-critical data and normal data traffic. The sporadic time-critical packets are triggered by high priority P_h nodes, and is bounded by strict deadlines. The normal data packets are periodically generated by low priority P_l nodes for process-related sensing and measurement. In the proposed network, all the deployed nodes are associated with the Access Point (AP)/Cluster Head (CH). The sensor nodes, including CH and the intermediate nodes have child-parent relationships, i.e., organized in the hierarchical model. Moreover, the proposed topology is considered as a static network over the time. The pictorial representations of the network scenario is presented in Fig. 4.3. The CH is responsible for distributing the clock information to different sensors. Similarly, the intermediate nodes are responsible for transferring the timestamp information from one hop to another.

In the proposed scheme, the P_l traffic transmission follows the conventional channel access scheme, and has been allocated a time slot with a duration T_s to transmit its data. The proposed superframe structure is shown in Fig. 4.4. The P_l node will initiate its transmission if the channel is found to be idle. Otherwise, the node will defer its transmission and wait until its next time slot. The Emergency Indication (EI) slot with duration T_{ei} can be used to prioritize P_h traffic over P_l. The packet transmission between P_h nodes and the CH is carried out in the form of consecutive transmission cycles. Each transmission cycle is composed of a time slot with a duration T_s and an EI slot with duration T_{ei}. The EI slot in each transmission cycle is followed by three phases; Clear Channel Assessment (CCA), Packet Schedule Phase (PSP),

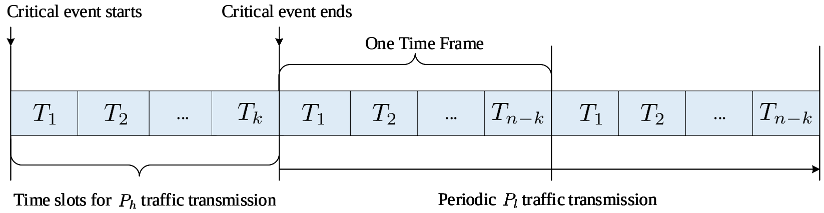

Figure 4.4: Superframe structure of the proposed priority-based protocol.

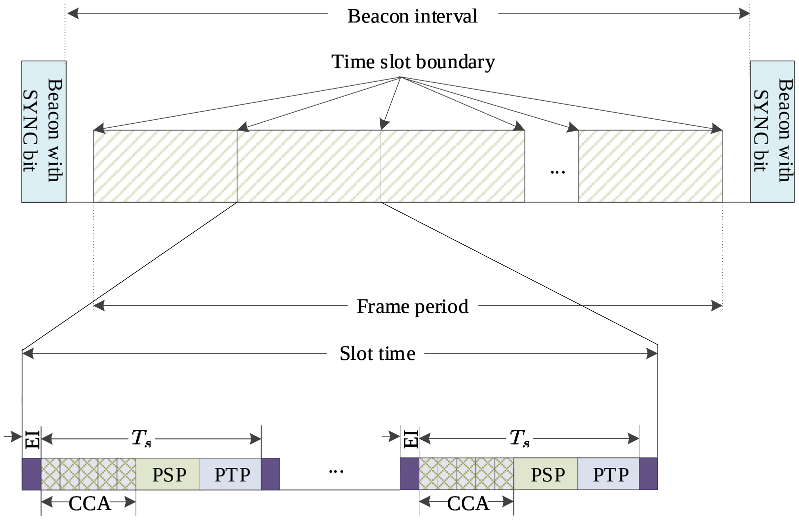

Figure 4.5: Illustration of the transmission cycle of P_h traffic in the proposed protocol.

and Packet Transmission Phase (PTP). The CCA is further divided into l dedicated subslots with duration T_l to send its channel access request. The transmission cycle of P_h is presented in Fig. 4.5. The deterministic channel access mechanism for the time critical nodes to complete its transmission within a delay bound, and clock synchronization and estimation process in the distributed network are detailed in subsections 4.4.1 and 4.4.2, respectively.

4.4.1 Deterministic Channel Access Mechanism

The non-deterministic random access delay is the major source of error in the clock synchronization process. The access delay mainly depends on the employed MAC layer protocol. The critical MAC layer challenge is to facilitate the deterministic channel access to a large number of sensor nodes with diverse service requirement and unique traffic characteristic. In this regard, we present the priority based deterministic channel access mechanism to reduce the channel access delay of the time-critical packets.

If node n_i has a P_h packet to send, it firstly transmits an indication signal within the EI time to take the slot from P_l nodes. After that, the node n_i asks for the guaranteed channel allocation for its P_h data transmission by sending a reservation packet to CH in the CCA subslot. The reservation request packet contains a binary payload, and corresponds to the decimal value d_i of the relative deadline (ms) of the P_h packet which needs to be delivered by n_i [71]. During the PSP, the activated P_h nodes are schedule based on the Earliest Due Date (EDD). Each node n_i will be scheduled for the channel access based on its d_i; such as the lowest d_i will gain the highest priority in the EDD schedule, and immediately access the channel [72]. This mechanism prioritizes the most urgent traffic for channel access to complete its transmission within a deadline bound.

Let n be the number of contending stations for the medium access in the given slot. In the case of saturation conditions, each station immediately transfers the packet after the completion of each successful transmission. However, due to the consecutive transmissions, each packet needs to wait for the random backoff interval before transmitting. The new arrival of the high priority packet P_h will immediately preempt lower priority packets currently being served on the queue, and get access to the services. During the channel access process, if the node detects a busy channel in present l subslots, then the node initiates a backoff stage. In the backoff stage, the node waits for a random number of time slots in the range of 0 to 2^{BE}, where BE is the backoff exponent and initial value is defined as $macMinBE$. For the delay estimation, we adapt the mathematical expression of CSMA/CA scheme of IEEE 802.15.4 presented in [34], [35]. Using this mathematical model, the expected service time in the network can be presented as [34]

$$E[S] = E[D] + T_{Tx} + 2T_{turn} + T_{ACK}, \tag{4.2}$$

where $E[D]$ represents the time period from the epoch that the packet just arrives at the queue to the epoch just before packet transmission, or discarded. The T_{ACK} and T_{Tx} are the transmission time of the acknowledgment and data packet respectively, and T_{turn} represents the turnaround time. The parameter $E[D]$ in the above expression depends on the CSMA/CA mechanism and is affected by several MAC parameters such as $macMaxBE$, $macMinBE$, CW, and *macMaxCSMABackoffs*. The expected value $E[D]$ can be derived as [34]

$$E[D] = \sum_{v=0}^{m} \alpha^{v}(1-\alpha)\left\{\sum_{i=0}^{v} \frac{CW_i - 1}{2}\sigma + (v+1)T_{CCA}\right\} + \alpha^{m+1}\left\{\sum_{i=0}^{m} \frac{CW_i - 1}{2}\sigma + (m+1)T_{CCA}\right\}, \quad (4.3)$$

where T_{CCA} is the time interval for performing CCA, α is the channel busy probability, and σ is the length of the backoff slot. The contention window size for the ith retry is expressed by; $CW_i = min\left\{2^i macMinBE, macMaxBE\right\}$. The default values of *macMinBE* and *macMaxBE* are 3 and 5, respectively [34]. The data packets are dropped or discarded after $m+1$ attempts at CCA, and consequently the packet loss rate is expressed by [34]

$$P_{loss} = \alpha^{m+1}. \quad (4.4)$$

Then, the channel busy probability α can be expressed in term of P_{loss} as [34]

$$\alpha = \frac{(n-1)(1-P_{loss})E[n_\tau](T_{CCA}+T_{Tx}+2T_{turn}+T_{ACK})}{\frac{1}{\lambda}+E[n_\tau]E[D]}, \quad (4.5)$$

where n is the number of sensor nodes, n_τ is the number of data packets served in a busy period of the queuing system, and $E[n_\tau]=\frac{1}{1-\rho}$ [47]. Therefore, by solving the above equations, *i.e.*, (4.3), (4.4), and (4.5), we can get the corresponding values of α, P_{loss}, and $E[D]$.

4.4.2 Distributed Clock Synchronization and Estimation Process

In this subsection, the clock synchronization and estimation process in distributed IoT network are presented in detail. Moreover, the effects of a number of nodes and hop counts on the overall precision of the clock synchronization are also described.

4.4.2.1 Clock Synchronization

The clocks may drift due to the variations in the oscillators, and the duration of the time intervals of different events will not be noticed identical among the IoT nodes. In general, the clock function of the ith node can be expressed as

$$C_i(t) = f_i t + \theta_i, \tag{4.6}$$

where the parameters f_i and θ_i are clock skew (i.e., frequency difference) and clock offset (i.e., phase difference), respectively. Similarly, the clock relationship between two nodes can be represented as

$$C_j(t) = f_{ij} C_i(t) + \theta_{ij}, \tag{4.7}$$

where f_{ij} and θ_{ij} are the relative clock skew and offset between node i and j, respectively. Let the clock of node i be the reference clock, then the function of the clock synchronization is to estimate the value of f_{ij} and θ_{ij} such that node j can adjust its timing information with reference node, when it is needed. The two nodes are perfectly synchronized when the value of $f_{ij} = 1$ and $\theta_{ij} = 0$. If the IoT network consists of n number nodes, then the global network-wide synchronization requires $C_i(t) = C_j(t)$ for all $i, j = 1, 2, 3, ..., n$ (i.e., all the relative clock skews and offsets should be estimated with respect to a reference clock). The graphical representation of the clock model of nodes is shown in Fig. 4.6.

The time synchronization error between clocks of node i and j at real time is given by $\mid C_i(t) - C_j(t) \mid$. Moreover, the average synchronization error of the overall network is the average of the time difference between every pair of nodes in the given network. For n number of nodes in the given network, the average synchronization error at real time t is given by

$$\zeta = \frac{2}{n(n-1)} \sum \mid C_i(t) - C_j(t) \mid, \forall i, i \neq j. \tag{4.8}$$

The main objective of the time synchronization scheme is to ensure that the average syn-

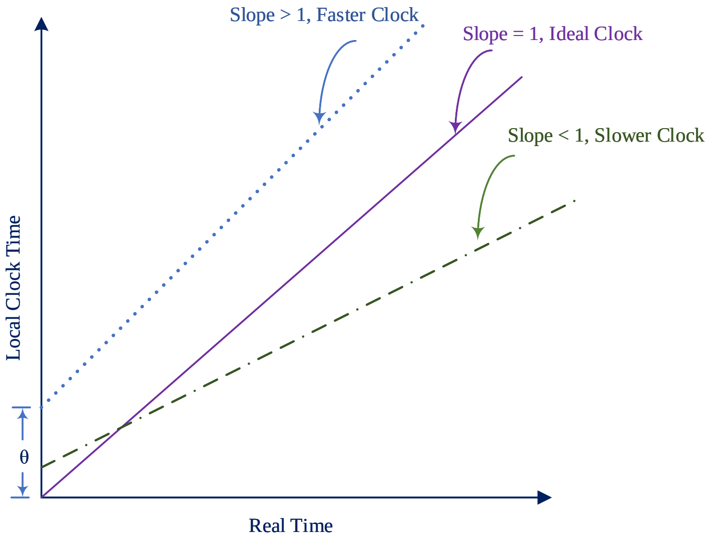

Figure 4.6: Clock model of IoT nodes.

chronization error at time t should be less than the maximum acceptable error. In the IoT network, the maximum acceptable error is application dependent and is in the range of μs. In general, the clock parameters are changes due to several factors such as temperature, pressure, voltage fluctuation, and hardware aging, thus, the network should perform periodic resynchronization mechanisms to adjust the clock parameters. The clock precision/accuracy is a measurement of the deviation of the error from the mean, and the stability measures the mean variation with respect to the above-mentioned factors [73].

4.4.2.2 Clock Estimation

The i reference node broadcasts ω_i number of packets for synchronization, and the nodes reply back with the η_j acknowledgment messages, where η_j is not necessarily equal to ω_i. The latency for line-of-sight (LOS) transmission between i reference node and the jth sensor node can be expressed as [74]

$$\tau_{i,j} = (f_j C_{i,j}^{(k)} + \theta_j) - (f_i t_i^{(k)} + \theta_i) + \varsigma_{i,j}^{(k)}, k = 1, 2, ..., K_i, \tag{4.9}$$

where $t_i^{(k)}$, $C_{i,j}^{(k)}$ are the transmission and reception timestamps of i reference node and jth sensor node, respectively, and $\varsigma_{i,j}^{(k)}$ is the aggregate estimation error on the timestamps. The transmission and reception timestamps recorded in the i reference node and jth sensor node can be presented as $T_i = [t_i^{(1)}, t_i^{(2)}, ..., t_i^{(K_i)}]^T \in \mathbb{R}^{K_i \times 1}$ and $R_{i,j} = [C_{i,j}^{(1)}, C_{i,j}^{(2)}, ..., C_{i,j}^{(K_i)}]^T \in \mathbb{R}^{K_i \times 1}$, where K_i is the number of transmission made by the i reference node. Similarly, the error vector can be expressed as $\varsigma_{i,j} = [\varsigma_{i,j}^{(1)}, \varsigma_{i,j}^{(2)}, ..., \varsigma_{i,j}^{(K_i)}]^T \in \mathbb{R}^{K_i \times 1}$.

For the neighborhood of n nodes, each synchronization packet piggybacks $n - 1$ previously received timestamps. The cycle of synchronization packet broadcasting is illustrated in Fig. 4.7, where $\Lambda_{i,j}$ denotes the jth synchronization packet of the node i, and $t_{i,j}^k$ denoted the reception timestamp at node k of the jth synchronization packet of node i. These timestamps are then used as samples to estimate relative skew and offset. The synchronization between node 1 and node 2 occur by gathering timestamp pairs ($t_{3,j}^1$, $t_{3,j}^2$) and ($t_{4,j}^1$, $t_{4,j}^2$). The timestamps obtained by each node can be arranged into the matrix of reception timestamps as [75]

$$T = \begin{bmatrix} * & t_{2,j}^1 & t_{3,j}^1 & t_{4,j}^1 \\ t_{1,j}^2 & * & t_{3,j}^2 & t_{4,j}^2 \\ t_{1,j}^3 & t_{2,j}^3 & * & t_{4,j}^3 \\ t_{1,j}^4 & t_{2,j}^4 & t_{3,j}^4 & * \end{bmatrix}, \tag{4.10}$$

where the * denotes the broadcasting node.

If u_k and v_k , $k \in \{1, ..., I\}$ denote the kth message reception timestamp of nodes 1 and 2, respectively, synchronization messages received by both nodes are use to construct timestamp samples. By using the maximum-likelihood estimation, we can obtained the resultant estimates for relative skew and offset using the timestamp values as

$$f_{ML} = \frac{\sum_{k=1}^{I} u_k \sum_{k=1}^{I} v_k - I \sum_{k=1}^{I} v_k u_k}{(\sum_{k=1}^{I} v_k)^2 - I \sum_{k=1}^{I} v_k^2}, \tag{4.11}$$

$$O_{ML} = \frac{1}{I}\Big(\sum_{k=1}^{I} u_k - \frac{\sum_{k=1}^{I} u_k \sum_{k=1}^{I} v_k - I \sum_{k=1}^{I} v_k u_k}{(\sum_{k=1}^{I} v_k)^2 - I \sum_{k=1}^{I} v_k^2} \sum_{k=1}^{I} v_k \Big). \tag{4.12}$$

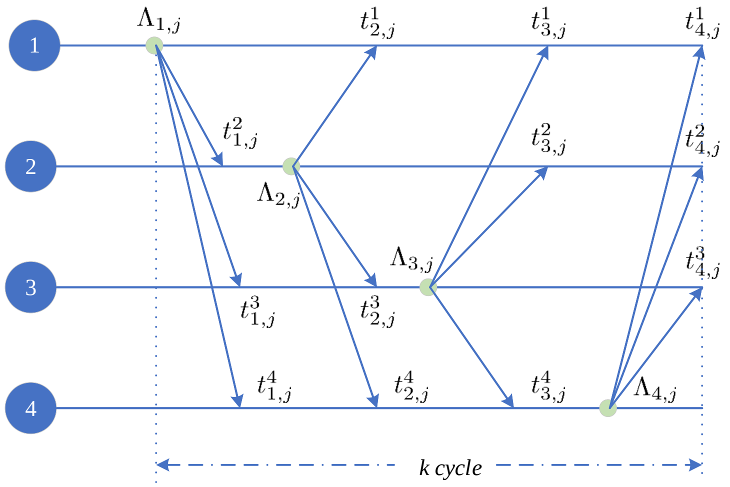

Figure 4.7: Synchronization packet broadcast during one cycle.

In the synchronization process, sensor nodes estimate the relative skew and offset with respect to neighbor nodes locally and independently. However, in a multi-hop scenario, the synchronization between end points is also required. The intermediate nodes may have to forward the synchronization parameters to the next hop to calculate the relative parameters. Consider the node n_1 needs to synchronize with the remote node n_r with the established link is $\{n_1, n_2, n_3, ..., n_r\}$. In this case, the intermediate nodes n_i have to estimate the local relative parameters and forward to the n_{i+1}. Let t_{n_i} be the n_i's clock time at instant t, and $f_{n_i \to n_j}$ and $\theta_{n_i \to n_j}$ represent the relative skew and offset between nodes n_i and n_j, respectively. The clock reading of the intermediate nodes in the link can be expressed as [70]

$$t_{n_i} = f_{n_{i+1} \to n_i} t_{n_{i+1}} + \theta_{n_{i+1} \to n_i}, i \in \{1, 2, ..., r-1\}. \tag{4.13}$$

Consequently, multi-hop estimators can be formulated as

$$f_{n_r \to n_1} = \prod_{i=1}^{r-1} f_{n_{i+1} \to n_i}, \tag{4.14}$$

$$\theta_{n_r \to n_1} = \sum_{i=2}^{r-1} \Big[\Big(\prod_{j=2}^{i} f_{n_j \to n_{j-1}} \Big) \theta_{n_{i+1} \to n_i} \Big] + \theta_{n_2 \to n_1}. \tag{4.15}$$

Algorithm 2 Clock adjustment mechanism of event critical applications

```
Event detection:Timer Started
  Set.timer(T)→ $C_i(t) = f_i t + \theta_i$;
  Send.msg();
Emergency Indication (EI): ON
Perform Priority Channel Access as presented in 4.4.1
Messages Received
  Gather Timestamps
  Perform Clock Estimation Procedure
Clock Adjustment
  Set.timer(T)→ $C_j(t) = f_{ij} C_i(t) + \theta_{ij}$;
  Send.msg();
  return;
```

The ever-increasing number of connected nodes in ultra-dense IoT networks and the dynamic traffic patterns increase the channel access delay and packet collision rate. In this regard, we utilized a priority based fast and efficient channel access scheme to enhanced the synchronization precision of the event critical IoT nodes. For multi-hop environments, intermediate nodes forward local synchronization parameters to the communicating nodes to allow them to calculate multi-hop parameters. In the presented network scenario, we assumed that each node had its own internal clock and ran independently from other nodes. The synchronization is guaranteed by estimating clock relative parameters of the reference node. In spite of continuous synchronization message broadcasting, the duty cycling scheme is enabled. The nodes are synchronized at the beginning of the time slot. In each duty cycle, the reference node broadcasts the beacons with a synchronization message. The synchronization message contains a timestamp, reference node ID, and sequence number. By incorporating the timestamp information within the beacon frames, the communication overhead can be significantly reduced as compared to the RBS like protocols [70]. We have also assumed that each node has the capability to communicate with at least one remaining node and to discover its one-hop neighbors.

The proposed scheme allows a network to determine a broadcast sequence by which nodes transmit and forward messages, and then conducts a network-wide synchronization based on received timestamp information. All the employed nodes are associated with coordinator/reference node. In the network initialization phase, all the nodes belonging to the same CH get the reference clock information during the synchronization period and contend to access the channel for data transfer. Moreover, to enhance the synchronization precision of the event critical nodes, the P_h nodes perform the clock estimation process with every event detection. The events are signaled by the hardware interrupts and depend on the specific platform [76]. The clock adjustment mechanism of event critical applications is presented in Algorithm 2.

4.5 Performance Analysis

In this section, we analyze and evaluate the performance of the proposed scheme in terms of the synchronization precision. The simulation parameters are presented in Table 4.2.

Table 4.2: Simulation Parameters

Parameters	Value
Max Backoff Exponent	5
Min Backoff Exponent	3
Max CSMA Backoff	4
MAC Frame Payload	800 bits
Queue Size	51 frames
Data Rate	19.2 kbps
ACK Size	88 bits
MAC Overhead	48 bits
σ	0.32 ms
T_{ACK}	0.352 ms
T_{Tx}	1.12 ms
T_{CCA}	0.25 ms
T_{turn}	0.192 ms

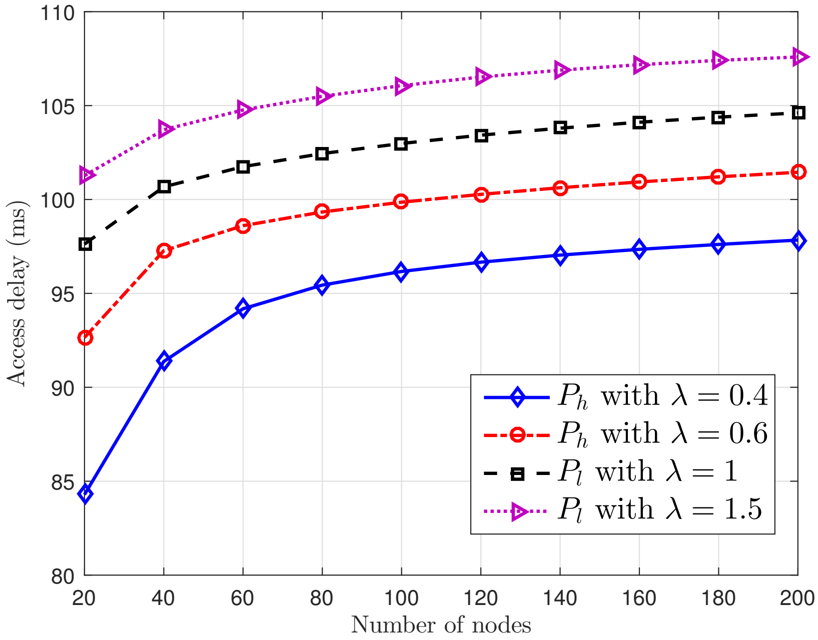

Figure 4.8: Performance evaluation of the proposed priority-based channel access mechanism in terms of the access delay.

4.5.1 Access Delay

Figure 4.8 shows the access delay of packets with different priority level versus the number of sensor nodes in IoT network. The packet arrival pattern follows the Poisson distribution with the arrival rate of λ packets per second. The average access delay of both low and high priority packets increases as the number of node increases, due to the contention mechanism and higher service time. The low priority packets have the longer access delay as compared to the high priority packets because the access time must accommodate the interrupts of the high priority event critical packets. In addition, due to the guaranteed channel access and deterministic packet scheduling in event based situation, the critical packets do not get any interruptions from low priority normal packets and hence, enhance the overall synchronization precision of the critical nodes.

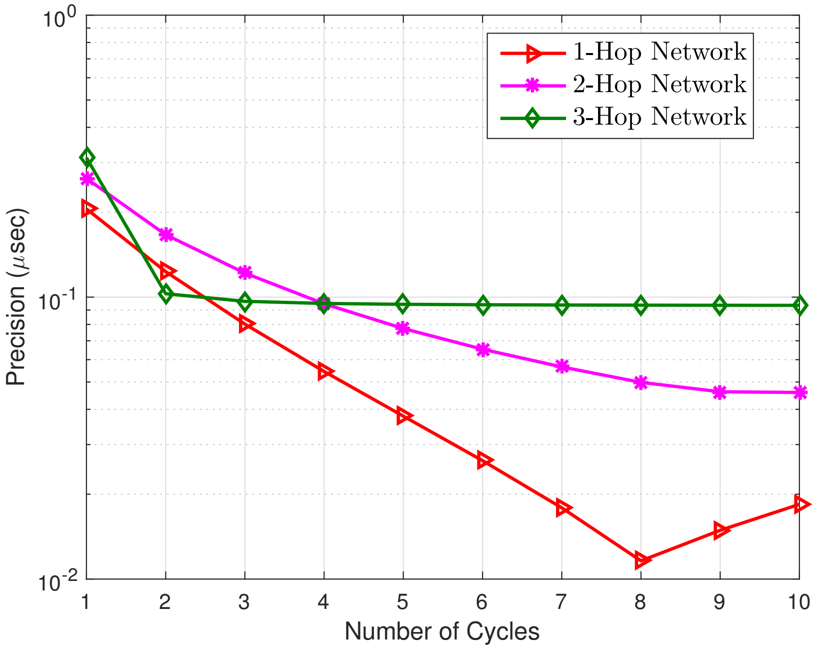

Figure 4.9: Performance comparison on synchronization precision versus number of cycles in multi-hop scenarios.

4.5.2 Effect of Network Hops

Figure 4.9 presents the effect of the network hops in the synchronization process. The nodes in each hop should run the synchronization process locally and independently as described in subsection 4.4. However, in the case of the multi-hop scenarios, the intermediate nodes have to forward the local synchronization parameters to the nodes in next hop, and allow them to estimate the relative parameters. From Fig. 4.9, it can be seen that the increase of synchronization samples/cycles considerably improves the precision with the number of hops. Also, the precision is found within the acceptable range even for the small number of synchronization samples/cycles. However, the synchronization precision decreases with the number of hops. The perceptive reason for the above mentioned trend is as following. The number of packet drops and collisions becomes more concurrent, and the probability of packet loss increases due to the retry limits and link constraints.

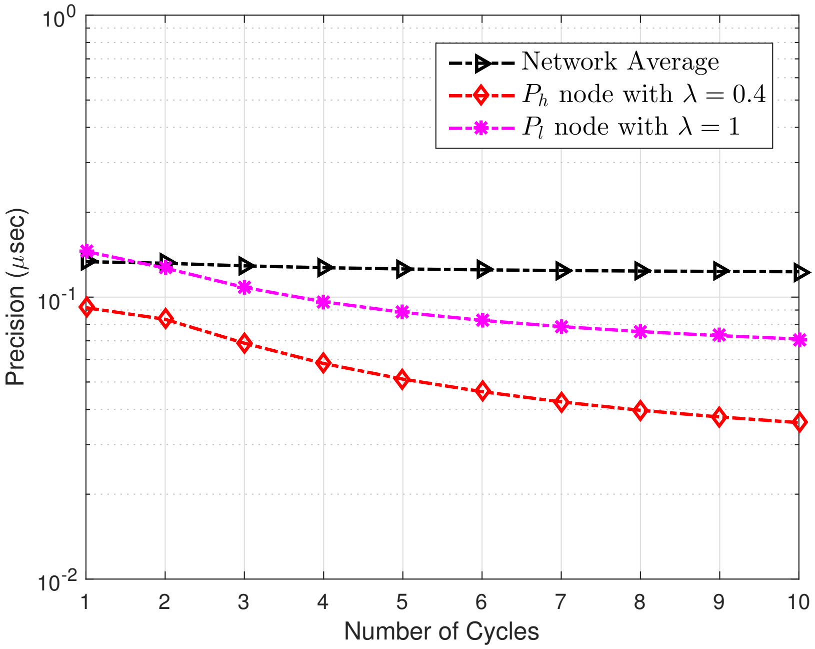

Figure 4.10: Performance comparison of critical nodes with normal nodes in terms of synchronization precision (Total number of sensor nodes = 100, and 30 percent of nodes are P_h).

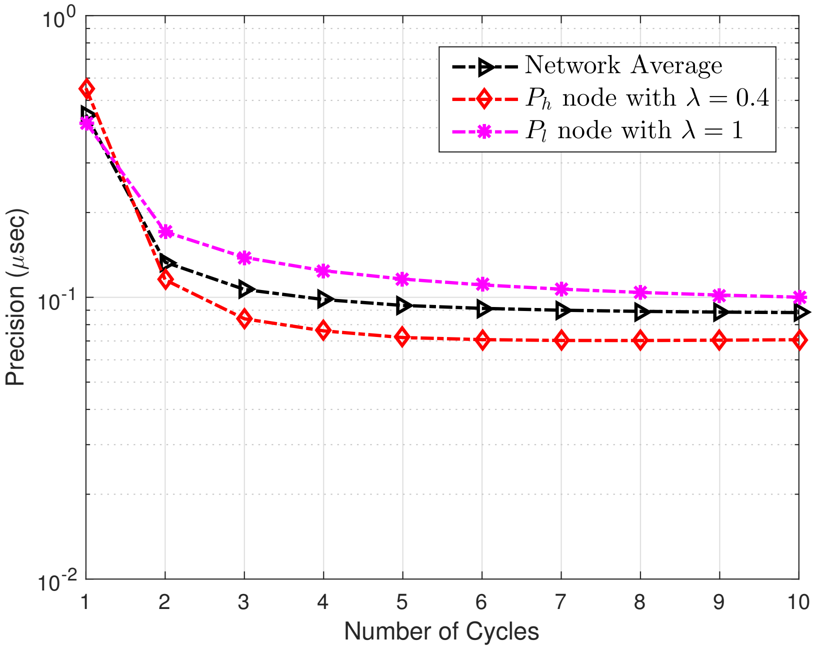

Figure 4.11: Performance comparison of critical nodes with normal nodes in terms of synchronization precision (Total number of sensor nodes = 200, and 30 percent of nodes are P_h).

4.5.3 Synchronization Precision

Figures 4.10 and 4.11 depict the synchronization precision versus number of cycles for a network with 100 and 200 nodes, respectively. In both scenarios, 30 percent of the network consists of the event critical high priority nodes (i.e., P_h nodes). From both figures, it is observed that the proposed scheme maintains the synchronization accuracy of the critical nodes much precise as compared to the normal nodes. The synchronization accuracy of the first scenario is more precise than the second case because of a smaller number of nodes in the network. The average precision deteriorates as the number of nodes increases due to the channel congestion and packet collisions. The proposed scheme guarantees predictable and timely channel access for the time-critical sensor nodes in wireless IoT networks. Moreover, due to the improved channel access mechanism and queuing policy, the network reliability of the event critical nodes is noted to be higher than that of normal nodes.

Figure 4.12 shows the performance of the synchronization precision of proposed mech-

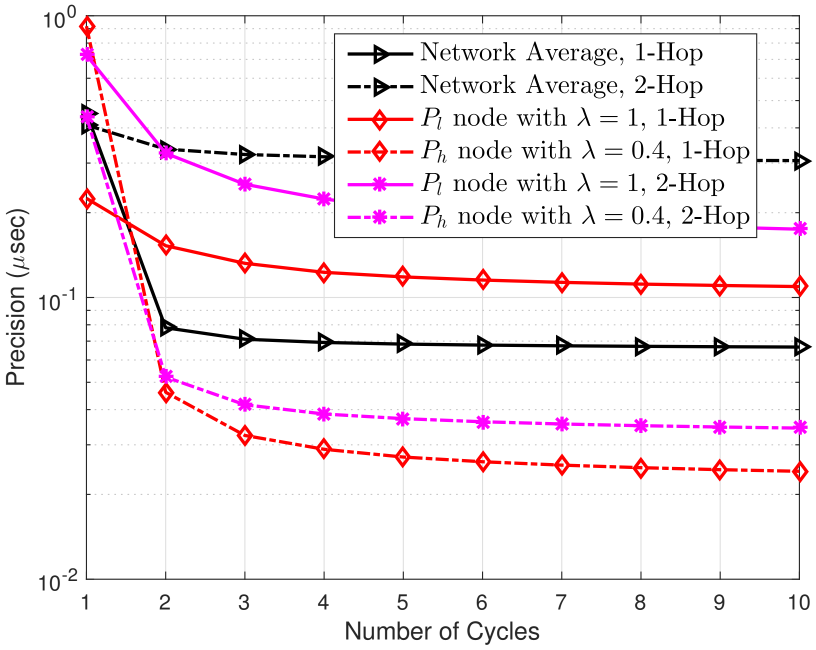

Figure 4.12: Performance comparison of critical nodes with normal nodes in terms of synchronization precision in multi-hop scenario (Total number of sensor nodes = 100, and 30 percent of nodes are P_h).

anism in multi-hop scenario. It is clearly observed that the synchronization precision deteriorates as the number of hops increases from 1 to 2. In the multi-hop scenario, the packet loss probability increases because of the link constraints, packet collisions, and retry limits. Moreover, the slot allocation decision coupled with predictable channel access mechanism contributes to bettering the synchronization accuracy of event critical sensor nodes as compared to normal nodes.

4.6 Chapter Summary

In distributed IoT network, precise clock synchronization mechanism is essential to perform the event driven sensing and control. The resource constrained IoT nodes are affected by packet delays, packet loss, and packet collisions. The packet collision rate increases significantly with node density. The problem of network synchronization becomes worse in IoT networks due to the uncertainties arising in the packet delivery process. In this chapter, an efficient clock synchronization scheme is proposed for the event critical applications in wireless IoT. The proposed scheme assigns time slots with high preference to the timestamp packets of critical nodes, and also guarantees the channel access in event based situations. Furthermore, the proposed scheme provides the deterministic packet scheduling, reduces the channel access delay, and enhances the synchronization precision. With the help of the simulation results, it has been shown that the proposed scheme substantially improves the synchronization precision of the event critical sensor nodes in comparison to the normal nodes. In our future work, we plan to implement the proposed algorithm in real hardware to analyze the performance of the proposed scheme in real-world IoT applications such as industrial automation and control sub-systems.

Chapter 5

Mobility and Location-aware Clustering Scheme for UAV Networks

5.1 Introduction

The Unmanned aerial vehicles (UAVs), also known as drones are considered as the enablers of many emerging applications in telecommunications, goods delivery, and surveillance [77]. The rapid development of wireless technologies such as low cost Wi-Fi modules, micro-computer, Global Position System (GPS), and sensors enables small UAVs to be extensively used in broader range of applications. However, a number of UAVs often have to be grouped as a collaborative swarm in carrying out critical missions due to the limited resource and capability of each UAV. The deployment of a large number of drones could bring some challenges such as collisions and interference, and subsequently affects the seamless operation of a UAV swarm. For the effective collaboration and cooperation among multiple UAVs, inter-UAV communication is critical to form Flying Ad-hoc Network (FANET). Moreover, UAV networks need a highly accurate location information with smaller interaction intervals due to the high mobility pattern in a multi-UAV environment.

In a FANET, one critical challenge is the effective management of a large number of mobile UAVs and various static ground stations. In overcoming this challenge, an extensive set of mini networks can be formed in an intelligent swarm. The self-organized network formation is an example of the intelligent cluster formation, where the UAVs are self-organized to reconnect themselves after a disruption in connections. Effective management of FANET is also directly related to the flying speed of UAVs, which are usually application dependent. The mobility of FANET is higher than that of Vehicular Ad-hoc Networks (VANETs) and Mobile Ad-hoc Networks (MANETs) [11]. The UAVs are highly mobile, with the speeds of 30 to 460 km/h [12].

The UAV mobility causes a significant impact on the link connectivity of UAV swarm networks. Effective management of UAV swarms and FANETs also relies on low latency communications. A wide variety of applications including surveillance, rescue operations, and disaster monitoring require minimal latency as the information needs to be transferred instantly. To control and minimize the communication latency, the concept of data prioritization has been developed. In addition, the priority-based routing protocol can be used to manage the Quality of Service (QoS) for various message types. Therefore, the implementation of the most suitable protocol is essential for minimizing the latency and improving the QoS of overall networks. In multi-UAV networks, the network may have several types of communication links such as UAV-to-UAV and UAV-to-ground link. The failure of a single UAV will disrupt the network stability and QoS requirements. Hence, the key features of mobile networks are reliability and survivability through redundancy.

The peer to peer connections are formed among the UAV swarms to maintain the coordination and collaboration, which can be effectively achieved by clustering /grouping [13]. For the homogeneous small-scale FANET, a single grouping is the best choice; however, for multi-purpose heterogeneous networks, there is a need for multi-cluster network. In this scenario, the Cluster Head (CH) is responsible for the inter-cluster communication as well as down-link communication. In the clustering process, the mobile UAVs are relocated in the cluster, where the position of CH is vertically projected on the centroid of the cluster. In the clustering process, CH selection and cluster formation schemes are very important to maintain the overall cluster structure. The clustering scheme enhances the overall QoS performance of the network such as network stability, throughput, and battery life [78].

The technical challenges in UAV networks are optimal deployment of UAVs, energy limitations, path planning, interference management, and stable wireless links. The optimal UAVs deployment and finding stable wireless links have great impacts on the network reliability and lifetime. Moreover, the packet drop rate and network latency are also dependent on the link stability between UAVs. The packet forwarding in UAV networks relies on the routing mechanisms applied in the MANETs. However, due to the frequent topology changes, high mobility, and unstable wireless links make the MANET protocols unreliable in UAV networks. The main contribution of this chapter is to propose a Mobility and Location-aware Stable Clustering (MLSC) scheme for randomly deployed UAVs network by incorporating the mobility and coverage probability. In this regard, we first present the coverage probability and the optimal number of CH UAVs can have to maximize the coverage area with the minimum transmit power in the given geographical area. Subsequently, we propose the *k*-means clustering mechanism to select stable CHs in optimal locations. Furthermore, we also present the cluster maintenance scheme with reference to the relative mobility and locations to enhance the stability of the cluster network.

The rest of this chapter is organized as follows. Section 5.2 summarizes the related works and provides the literature review in the area of clustering schemes in UAV networks. In Section 5.3, the proposed location and mobility aware clustering scheme is described in detail. In Section 5.4, the performance of the proposed scheme is evaluated via simulations. Section 5.5, concludes the chapter.

5.2 Related Works

The clustering is an efficient network management scheme that can improve the overall performance of the ad-hoc UAVs network by dividing the complex network into the number of clusters. The clustering in UAV network provides several benefits such as reliability, scalability, fault tolerance, energy efficiency, latency minimization, coverage maximization, and stable connectivity. The literature of existing clustering algorithms are mainly classified into two categories [79]: (i) probabilistic clustering and (ii) deterministic clustering. The main objective of the probabilistic cluster algorithm is to find the best routing route by making the network lifetime longer. The probabilistic clustering algorithms can further be classified into three categories: (i) dynamic clustering, (ii) bio-inspired clustering, and (iii) hybrid clustering.

The UAV Routing Protocol (URP) [80] and UAV-based Linear Sensor Networks (ULSNs) [81] are examples of dynamic clustering algorithm. The URP and ULSN can effectively reduce the resource requirements of the network, and also improve the network lifetime. However, these algorithms are mainly designed for the Wireless Sensor Networks (WSNs) with the single UAV. In [82], the authors proposed the Energy-Aware Link-based Clustering (EALC) algorithm to address the problems related to the inefficient routing and UAV flight time. The author used a k-means algorithm to enhance the network lifetime by finding optimal cluster. Similarly, in [83] authors presented the Bio-Inspired Mobility Prediction Clustering (BIMPC) algorithm for the cluster formation and maintenance of large scale UAV networks. However, in both schemes, the authors did not consider the randomness and high mobility patterns of the UAVs. Moreover, the bio-inspired based Ant Colony-Bee Colony (AC-BC) scheme, Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Grey Wolf Optimization (GWO) are also used to perform the clustering in UAV networks [84, 85]. Nevertheless, these schemes did not consider the coverage probability and the optimal number of CH UAVs can have to maximize the coverage area with the minimum transmit power in the given geographical area. To solve the issues related to the connectivity, coverage and energy consumption, the authors in [86] proposed the Received Signal Strength Indicator (RSSI) based Hybrid and Energy-Efficient Distributed (rHEED) based on HEED algorithm [79]. The rHEED scheme utilized the RSSI from the received from UAV, and also consider the residual energy of the node to elect the CH. This scheme provides the balanced and stable cluster. However, this scheme is proposed by considering a single UAV based WSN, and is not suitable for the UAV networks.

The Ad-hoc On-Demand Distance Vector (AODV) protocol is also widely used in UAV networks. However, due to the dynamic link connections, it suffers from the network overhead and latency issues [87]. In [88], authors evaluated the performance of the Optimized Link State Routing (OLSR) protocol in UAV network comprising of ground stations and two UAVs, and conclude that the OLSR is unreliable in UAV networks due to the rapid topology changes. In the case of the deterministic clustering algorithms, the CHs are elected based on the information exchanged by the neighboring UAVs. The common metrics used to elect the CHs are centrality, proximity, randomness, mobility, and residual energy. In [89], the authors present the scalable multiple target tracking system by applying Density-based Spatial Clustering Applications with Noise (DBSCAN) algorithm. The locations of the mobile target are estimated by using the extended Kalman filters. The main advantages of the proposed scheme are the path planner and optimal sensor manager to get the geolocations of targets within the cluster. The authors in [90] presented the Mobility Predication Clustering Algorithm (MPCA) based on the dictionary trie structure prediction algorithm and link expiration time mobility model. The proposed scheme is very useful to manage the stability of the network. However, due to the high mobile environment, the shape of the cluster structure changes rapidly, and a large amount of packet overhead will be introduced to maintain the stability of the cluster. The geographical-based routing protocol is presented in [91], and the authors considered mobility, direction, and velocity of UAVs to estimated the UAV link lifetime. Nevertheless, this work did not consider the coverage probability of UAV in a given geographical area. The work in [92] investigated the optimal movement and deployment area of the UAV to support the downlink wireless communications. However, the proposed scheme was limited to the single UAV and only considered for the downlink. In addition, the existing schemes did not consider the joint impact of the coverage probability and mobility of UAVs in cluster formation and maintenance mechanisms.

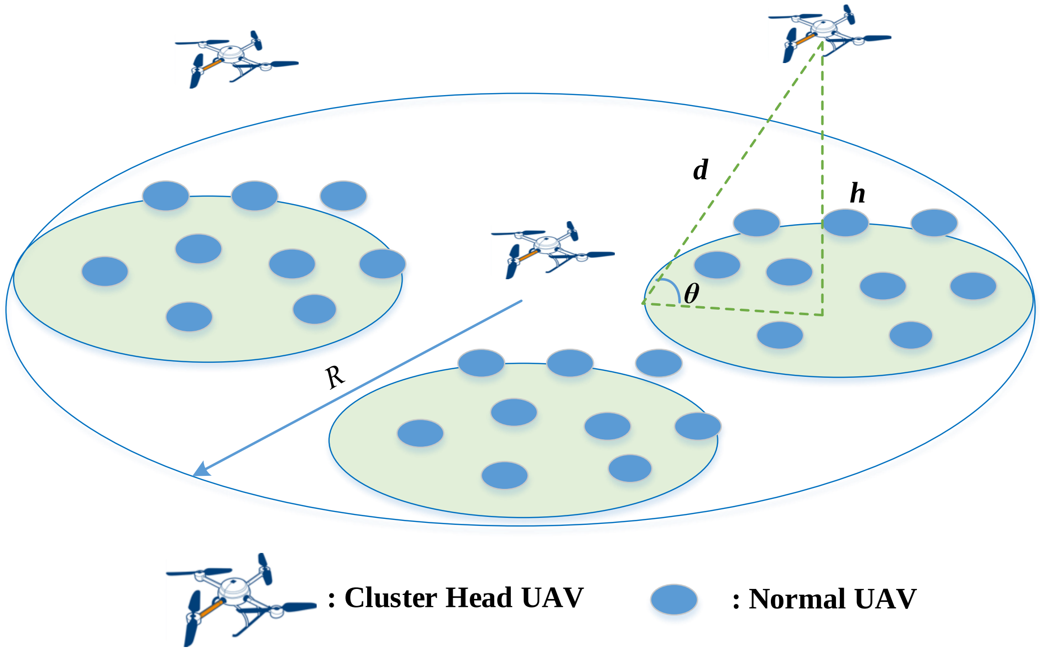

Figure 5.1: The optimal deployment of CH UAVs to maximize the coverage probability.

5.3 System Model and Proposed Schemes

In this section, the location and mobility aware clustering mechanism is described in detail. Firstly, the optimal deployment of CH UAVs to maximize the coverage probability is presented. This model studies the relationship between the size of the cluster and maximum coverage probability in the network to find the optimal cluster size to minimize the number of transmissions. Secondly, the proposed distance based *k*-means clustering algorithm and cluster maintenance scheme are described in detail.

5.3.1 Efficient Deployment of CH UAVs

In this subsection, the optimal deployment of the CH UAVs in order to maximize the coverage area with the minimum transmit power is investigated. For the given target geographical area, the number of CH UAVs are equipped with the single antenna. The main objective of this scheme is to maximize the coverage performance by ensuring the coverage fields of UAVs do not overlap. The deployment model with a circular geographical area of radius R is shown in Fig. 5.1, where K CH UAVs must be deployed to provide the wireless coverage for the normal UAVs. The UAVs are assumed to have same transmit power. The CH UAVs' antenna gain can be approximated as [93]

$$G = \begin{cases} G_{\mathrm{3dB}}, & \frac{-\theta_B}{2} \leq \varphi \leq \frac{\theta_B}{2}, \\ g(\varphi), & \text{otherwise}, \end{cases} \tag{5.1}$$

where φ is the sector angle, $G_{3dB} \approx \frac{29000}{\theta_B^2}$ with θ_B in degrees is a main lobe gain, and $g(\varphi)$ is the antenna gain outside of the main lobe. The common approach for a channel modeling is to consider the Line-of-Sight (LoS) and Non-Line-of-Sight (NLoS) links between CH UAVs and normal UAVs. Each link has a distinct probability of occurrence which depends on the elevation angle, environment, and relative location of the CH UAVs and the normal UAVs. The shadowing and blockage loss for the NLoS links are higher as compared to the LoS Links. The received signal power at UAVs can be given as [94, 95]

$$P_{r,j}(dB) = \begin{cases} P_t + G_{\mathrm{3dB}} - L_{dB} - \psi_{\mathrm{LoS}}, & \text{for LoS link}, \\ P_t + G_{\mathrm{3dB}} - L_{dB} - \psi_{\mathrm{NLoS}}, & \text{for NLoS link}, \end{cases} \tag{5.2}$$

where $P_{r,j}$ is the received signal power, P_t is the CH UAV's transmit power, and G_{3dB} is the antenna gain in dB.

Also, the path loss L_{dB} is expressed as

$$L_{dB} = 10n\log\left(\frac{4\pi f_c d_j}{c}\right), \tag{5.3}$$

where f_c is the carrier frequency, c is the speed of light, d_j is the distance between CH UAV and normal UAVs, and $n \geq 2$ is the path exponent. Similarly , $\psi_{\mathrm{LoS}} \sim N(\mu_{\mathrm{LoS}}, \sigma^2_{\mathrm{LoS}})$ and $\psi_{\mathrm{NLoS}} \sim N(\mu_{\mathrm{NLoS}}, \sigma^2_{\mathrm{NLoS}})$ are shadow fading with normal distribution in dB scale for LoS and NLoS links. The variance can be given as

$$\begin{aligned}\sigma_{\text{LoS}}(\theta_j) =& k_1 \exp(-k_2\theta_j),\\ \sigma_{\text{NLoS}}(\theta_j) =& g_1 \exp(-g_2\theta_j),\end{aligned} \tag{5.4}$$

where $\theta_j = \sin^{-1}(h/d_j)$ is the elevation angle (in radians) between CH-UAV and normal UAVs, k and g are constants, and depends on the environment.

Finally, the LoS probability is calculated as

$$P_{\text{LoS},j} = \alpha\left(\frac{180}{\pi}\theta_j - 15\right)^{\gamma}, \tag{5.5}$$

where α and γ are constant values reflecting the environment impact. Hence, the NLoS probability is given as [96, 97]

$$P_{\text{NLoS},j} = 1 - P_{\text{LoS},j}. \tag{5.6}$$

Our main goal is to provide the wireless coverage to the largest possible number of UAVs with minimum number of CH UAVs. The number of CH UAVs depends on the expected coverage in geographical area and the number of available normal UAVs. In this scenario, the number of normal UAVs is fixed to N and the number of CH UAVs is K. The main objective is to determine the optimal number of CH UAVs to achieve full coverage to N users. Let γ_{ij} is the Signal to Interference plus Noise Ratio (SINR) between UAVs i and j, and I_{ij} be an indicator of whether or not UAV i is connected to UAV j such that [98]:

$$I_{ij} = \begin{cases} 1 & \text{if } j = \arg\max\limits_{j \in \mathcal{M}} \gamma_{ij} \text{ and } \gamma_{ij} \geq \gamma_{\text{th}}, \\ 0 & \text{if otherwise.} \end{cases} \tag{5.7}$$

The problem can then be formulated as:

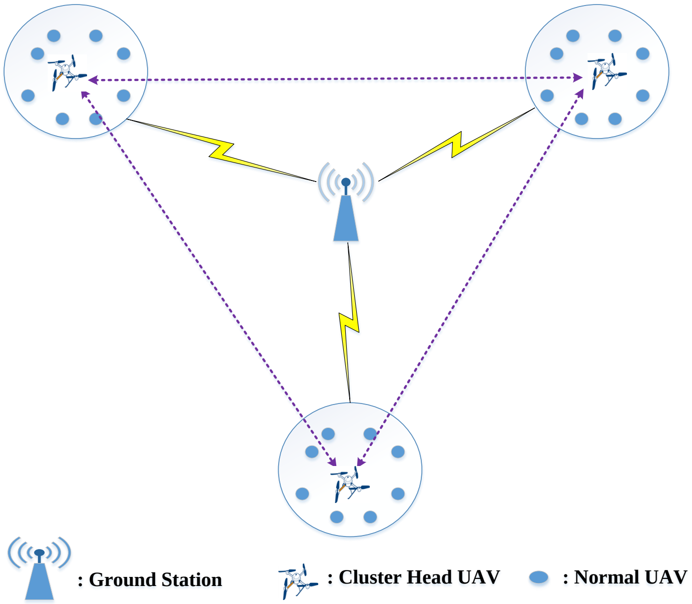

Figure 5.2: Location and mobility aware clustering in UAV networks. CHs are responsible to forward the packets to the ground station/sink.

$$\min_{\mathcal{K}} \sum_{j\in\mathcal{K}} \sum_{i\in\mathcal{N}} I_{ij} \tag{5.8}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{K}} I_{ij} = 1, \forall i \in \mathcal{N}, \tag{5.9}$$

$$\sum_{i\in\mathcal{N}} I_{ij} = N. \tag{5.10}$$

The first constraint ensures that every UAV is connected to only one CH and the second constraint ensures that all the UAVs are connected to CHs. This model ensures the optimal number of CH for a given number of UAVs in the field.

5.3.2 Location Based Cluster Formation

In this subsection, we present the clustering mechanism by using k-means clustering algorithm. Based on Section 5.3.1 analysis, we calculate the optimal number of a cluster for a given N

number of UAVs in the field. The proposed scheme has two major steps; (i) elect K CHs from the set of N UAVs and divide into the K cluster with optimal size of the each cluster N_k as $N_k = \left[\frac{N}{K}\right]$, and (ii) formulate the backbone route to connect all the CHs to the sink (i.e., ground station). The UAV network can be modeled by $G = \langle U, D \rangle$, where U consists of the sink node u_0 and N UAVs. If the two UAVs are in communication range of each other, then there is a link between them. The sink/ground station has the full knowledge of the network topology. The sink is responsible for formulations of the clusters, elections of the CH for each cluster, and constructions of the backbone routing tree. The backbone routing tree connects all CHs and the sink.

The centralized clustering mechanism is presented in Algorithm 3. The main objective of the k-means clustering algorithm is to perform the clustering of given N number of UAVs to K different clusters/groups. The important factors for the efficient clustering are to maximize the coverage probability and to determine the optimal size of the cluster. The size of the cluster affects the number of transmissions in the network. If the cluster size is large, the number of transmissions required to collect the data from member UAVs to the CH UAV will be very high and, thereby, affects the network performance. Similarly, if the cluster size is too small, the number of clusters will increase, and the data transmissions from all CH UAVs to the ground

Algorithm 3 Centralized clustering mechanism

```
Input: Number of cluster $K$ for all $n \in N$
Output: CHs and corresponding cluster members $CM_i$
Start
Remaining UAVs← All UAVs
while Remaining UAVs ($R_{UAVs}$)!=0 do
    Cluster the UAV network based on location using (5.11)-(5.13)
    $CH_i \leftarrow$ UAV having minimum distances from other UAVs
    $CM_i \leftarrow$ All UAVs in $CH_i$ transmission range
    $R_{UAVs} \leftarrow R_{UAVs} - CMs$
Return $CH_i$ and $CM_i s$
End
```

station will be very large. This, ultimately, leads to the degradation of the QoS performance of the network. From the theoretical analysis in the previous section, we can calculate the optimal number of CH UAVs for a given number of UAVs distributed in a field. To form the clusters, the CH UAVs are selected first, and Euclidian distance is calculated from each member UAV to all CH UAVs and finally, allocated to the nearest CH UAV. The main goal is to minimize the Euclidean distance of each member UAV to the closest CH UAV. The cost function to find the optimum μ_j can be defined as [99]

$$C_{n,j} = \sum_{n=1}^{N} \sum_{j=1}^{J} r_{n,j} ||x_n - \mu_j||^2, \tag{5.11}$$

where $r_{n,j}$ is defined as

$$r_{n,j} = \begin{cases} 1, & \text{if } j = \arg\min_i ||x_n - \mu_i||^2 \\ 0, & \text{otherwise.} \end{cases} \tag{5.12}$$

To minimize the cost function, first take the derivative with respect to μ_j and set to zero, which gives

$$\mu_j = \frac{\sum_{n=1}^{N} r_{n,j} x_n}{\sum_{n=1}^{N} r_{n,j}}. \tag{5.13}$$

In addition, we also used the Minimum Spanning Tree (MST) algorithm [100, 101] to formulate the backbone route. The backbone tree construction mechanism is presented in Algorithm 4. The backbone route connects the all CHs and sink/ground station. A set S of CHs is obtained from the above clustering scheme, and we introduced a graph $G_{CH} = \langle U_{CH}, D_{CH} \rangle$, where U_{CH} consists of the sink node u_0 and set S of CHs. The distance D_{CH} is the shortest path between (CH_i, CH_j) in G. Then, we calculate the MST of the G_{CH}, and formulate the routing tree between all CHs and sink. In the auxiliary graph $G_{CH} = \langle U_{CH}, D_{CH} \rangle$, and each CH UAV in has an edge v to the each of the UAVs in its neighboring cluster. The distance of each edge (u, v) in E is taken in non decreasing order such that the total distance from the member UAVs in A to their nearest CH UAV is minimized. The computational complexity for

Algorithm 4 Construction of the backbone tree for data transmission from CHs to sink

```
Input: Sink node $u_0$, set $S$ of CHs, and distance $D_{CH}$ between CHs
Output: List of edges $A$ in MST to connect all CHs and the Sink node
Initialize: $G_{CH} = \langle U_{CH}, D_{CH} \rangle$
A ← ∅
for each vertex $v \in V[U_{CH}]$ do
    MAKE-SET($v$)
Sort the edge nodes of $D_{CH}$ into non decreasing order by locations
for each edge $(u, v) \in E$, taken in non decreasing order by minimum distance do
    if FIND-SET($u$) ≠ FIND-SET($v$) then
        A← $A \cup (u, v)$
        UNION $(u, v)$
Return A
```

k-means clustering is on the order of $O(K * N * I * D)$, where K is the number of clusters, N is the number of UAVs, I is the number of iteration, and D is the dimension or number of the attributes [102]. Similarly, the computational complexity of the MST algorithm is $O(ELogv)$, where E is the number of edges, and v is the vertices in the graph.

5.3.3 Distributed UAV Network Implementation

In this subsection, the distributed UAV network implementation of the proposed clustering algorithm scheme is presented. The UAV knows its speed and the geographical information. The location information and speed can be obtained from the attached GPS or by implementing the localization techniques. In addition, the sink knows the coverage area of the field, but does not know the location of the deployed UAVs.

The CH broadcasts an *advertisement* message to the UAVs in the cluster field to join the specified cluster. The *advertisement* message carries the information such as ID and location of CH, and the number of hop count. After receiving the *advertisement* message, the UAV updates the CH information if the hop count of the message is smaller than the pre-recorded value from same CH, and further broadcasts the message to its neighbor UAVs. After completion of

CH advertisement, each UAV decides to join corresponding cluster based on the distance and number of hops to each CH.

The backbone route can be constructed in a distributed manner to connect all CH UAVs and the sink. The CHs can share their locations information by broadcasting the *advertisement* messages. The sink broadcast the central information to the UAVs through the respective CH. The distributed method of an approximate MST algorithm is used to construct the backbone network. For each CH, it elects the CH that has minimum number of hops from the set of CHs as its parent CH in the backbone route. After completing the backbone tree, each CH can have the information about neighbor CHs in the backbone tree.

5.3.4 Cluster Maintenance

The cluster's stability rapidly degrades in a highly mobile environment. Hence, the relative speed S_k, defined for each UAV represents the good measure for the stability of a cluster. This metric can be evaluated as the average difference in velocities v between the CH UAV k and all N neighboring UAVs within its range, i.e. those belonging to the set Φ_k. Moreover, the value is normalized to be within the range of [0, 1]. The relative mobility can be expressed as [103]

$$\mathrm{S}_k = \frac{\sum_{n=1}^{N} |v_k - v_n|}{N \cdot max\{\Omega_k\}}, \tag{5.14}$$

where the normalizing factor is the maximum value of the set Ω_k, and can be expressed as

$$\Omega_k = \{|v_k - v_n| \,|(v_k, v_n) > 0; \forall n \in \Phi_k\}. \tag{5.15}$$

Another metric that can be used to determine a stable CH UAV is its relative position to the neighbors. A smaller normalized relative mean distance ∂_k indicates that the neighboring UAVs are closer to the potential CH UAV. Consequently, the mean relative distance ∂_k of UAV k is defined as the mean Euclidean distance. Furthermore, by normalizing with the maximum value of the set Z_k as in (5.16) makes S_k and ∂_k comparable.

$$\partial_k = \frac{\sum_{n=1}^{N} \sqrt{[\Delta x_{k,n}]^2 + [\Delta y_{k,n}]^2 + [\Delta z_{k,n}]^2}}{N \cdot max\{Z_k\}}, \tag{5.16}$$

where the normalizing factor is the maximum value of the set Z_k, which is composed of all the Euclidean distances between UAVs.

Similarly, the normalized relative mean distance between the ground station/sink and the inter-cluster UAVs can be expressed as

$$\mathscr{D}_k = \frac{\sum_{n=1}^{N} \sqrt{[\Delta x_{k,0}]^2 + [\Delta y_{k,0}]^2 + [\Delta z_{k,0}]^2}}{N \cdot max\{\varphi_k\}}, \tag{5.17}$$

where φ_k is composed of all the Euclidean distance between sink and inter-cluster UAVs.

Finally, the CH selection index is evaluated as the sum of the normalized values of the mean relative speed and distances as

$$\xi_k = S_k + \partial_k + \mathscr{D}_k, \tag{5.18}$$

which always fall in the range [0, 3]. Upon periodical exchange of the packets amongst all the UAVs in the cluster, the kth UAV can record a list of all CH selection indexes ξ belonging to every nth UAV in its neighbor's set Φ_k. The set of all ξ for every neighbor's set Φ_k can be defined as [103]

$$\Psi_k = \{\xi_n | \forall n \in \Phi_k\}. \tag{5.19}$$

To make the network stable and reliable, we have to maintain the cluster structure. The cluster maintenance and backup cluster election procedure is presented in Algorithm 5. The proactive backup cluster head CH_{bkp} scheme is introduced to fulfill the CH position, if current CH is resigned or away from the network. The choice of stable CH_{bkp} is assigned based on the selection index ξ. We also defined a set of all UAVs belong to the same cluster and CH as $\varnothing_i$. The CH keeps all the information of its CMs and knowledge of neighbours set Φ_k of every kth CM. The UAV k with ID ς_k will be elected CH if the selection index ξ_k is found to be smaller

Algorithm 5 Cluster maintenance and backup cluster head selection

```
for each $CH_i$ in $\varnothing_i$ do
    $CH_i$ assigns the $CH^i_{bkp}$ using (5.20)
if $CH_i$ leaves network then
    $CH_{bkp} \leftarrow CH_i$
if $CH_i$ is in the coverage zone of another $CH_j$ then
    if $CH^i_{bkp}$ is not in coverage zone of another $CH_j$ then
        $CH_{bkp} \leftarrow CH_i$
    else
        Merge cluster $\varnothing_i$ and $\varnothing_j$
if $CM_i$ is not in coverage zone of $CH_i$ then
    go to CH election Algorithm 3
```

than ξ_n and can be expressed as

$$CH_{bkp} = \{\varsigma_k | \xi(\varsigma_k) \leq min\{\Psi_k\}\}. \tag{5.20}$$

Table 5.1: Simulation Parameters

Parameters	Value
Simulation area	2000 m*2000 m*2000 m
Simulation round	2000
Number of ground station	1
Number of UAVs	20-140
MAC protocol and frequency	IEEE 802.11, 2.4 GHz
Transmission range	250-300 m
Traffic type	CBR
Packet size	20-500 kB
CBR rate	2 Mbps
UAV speed	10-30 m/s
Mobility model	Gauss-Markov mobility model

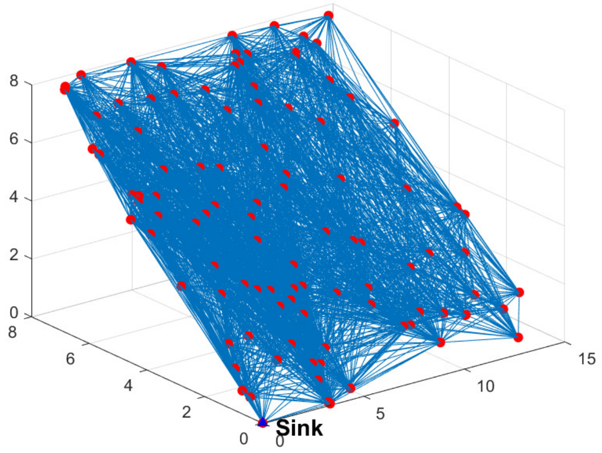

Figure 5.3: UAV node connectivity without clustering (Axis units are x100 meter). If all the UAVs trying to communicate with each other, then the network overhead will increase exponentially.

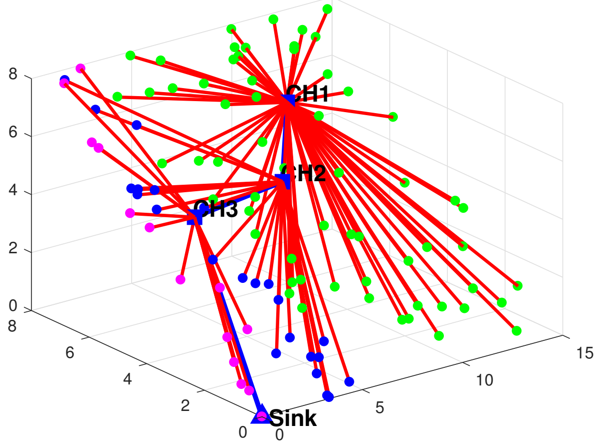

Figure 5.4: UAV network after clustering (Axis units are x100 meter). UAVs in each cluster transmit the packet to the CH. CHs are responsible to forward the packets to the ground station/sink.

5.4 Performance Analysis

In this section, the performance of the proposed scheme is analyzed and evaluated by using the MATLAB software. The simulation parameters are presented in Table 5.1. The IEEE 802.11 radio standard [15] operating in the 2.4 GHz frequency band for wireless communication is used. First, the optimal number of CH is determined to enhance the coverage probability, which also enhances the network performance by reducing the number of network overheads. Based on the analytic result, the proposed MLSC scheme is implemented in the UAV network. The deployed UAV network and the network after the proposed clustering algorithm are shown in Fig. 5.3 and Fig. 5.4, respectively.

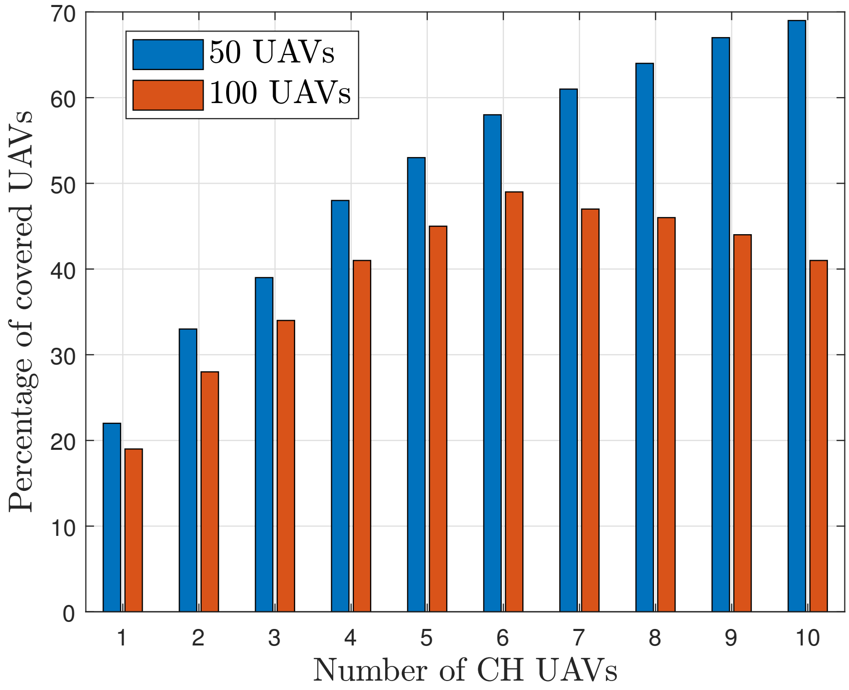

Figure 5.5: The impacts of a number of CHs on the coverage performance for various network size (i.e., for UAVs network size of 50 and 100).

5.4.1 Impacts of a Number of Cluster Heads

Figure 5.5 presents the impact of the CH UAVs on the coverage performance in various network sizes (i.e., no of UAVs) in the given deployment scenario. It is clearly shown that the coverage performance decreases as network size increases. There is a trade-off in deploying more CH UAVs to provide the optimal coverage. By increasing the CH UAVs (or the number of clusters), the coverage can be improved. However, by increasing the number of clusters, the aggregated interference increases which reduces the SINR value. For instance, the optimal number of CH UAVs for serving 100 UAVs is 6.

5.4.2 Impacts of a Number of Clusters

Figure 5.6 represents the impact of the cluster size on the performance of the proposed scheme. The number of UAV is set 100 and 50, and the number of clusters K varies from 1 to 15. The

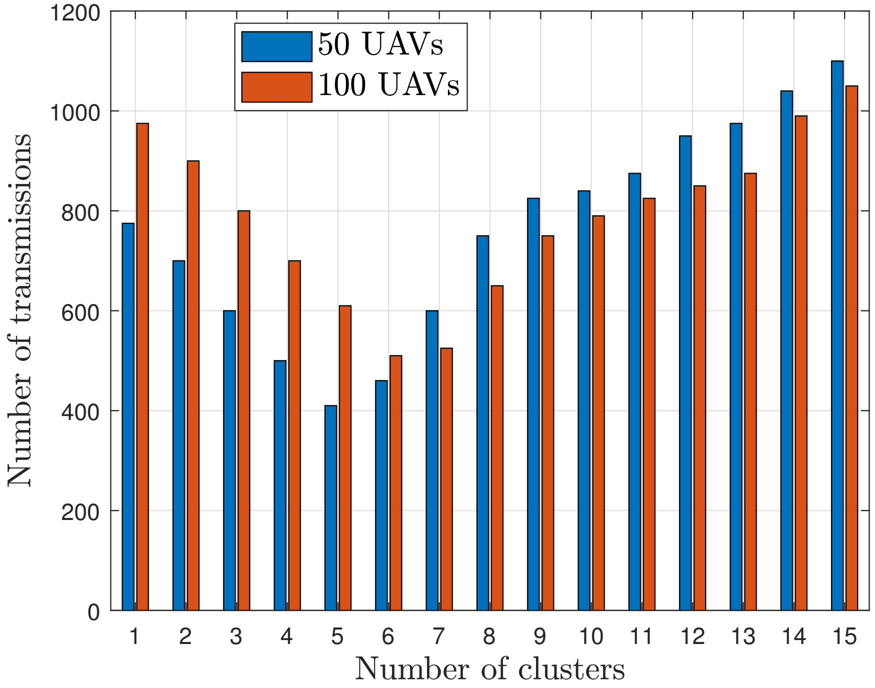

Figure 5.6: The impacts of a number of clusters on the data transmission for UAV networks of size 50 and 100.

main objective of the proposed scheme is to minimize the number of transmission in the cluster network, which is sum of the intra cluster transmission and the inter cluster transmission. From the Fig. 5.6, it is seen that the number of transmissions decreases as the number of cluster increases until N_{ck} reaches certain value, afterwards increases of N_{ck} would lead to increase of transmissions.

5.4.3 Normalized Routing Overhead

Figure 5.7 depicts the normalized routing overhead for various UAV velocities. The mobility of UAV causes high route request rate and increases the control packet overhead. The control packet overhead surpasses the data rate, and also enhances the packet drop and network latency. The normalized routing overhead of proposed MLSC and AODV increases with increased UAV velocity in both cases. The routing overhead in AODV is very high as compared to the proposed MLSC scheme because a large amount of time is required to find a path in the

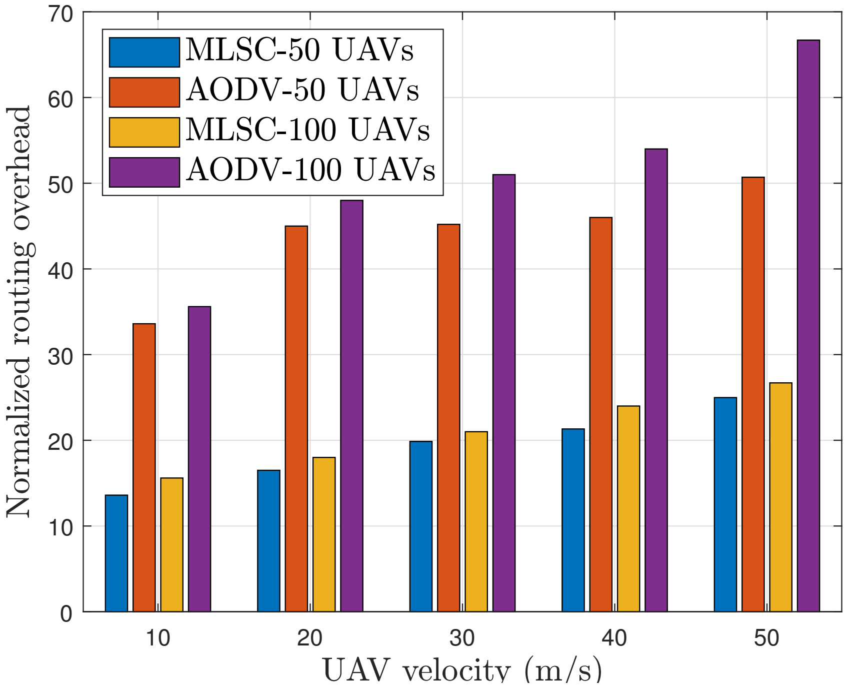

Figure 5.7: Performance comparison of the proposed MLSC scheme with conventional AODV protocol in terms of normalized routing overhead versus UAV velocity.

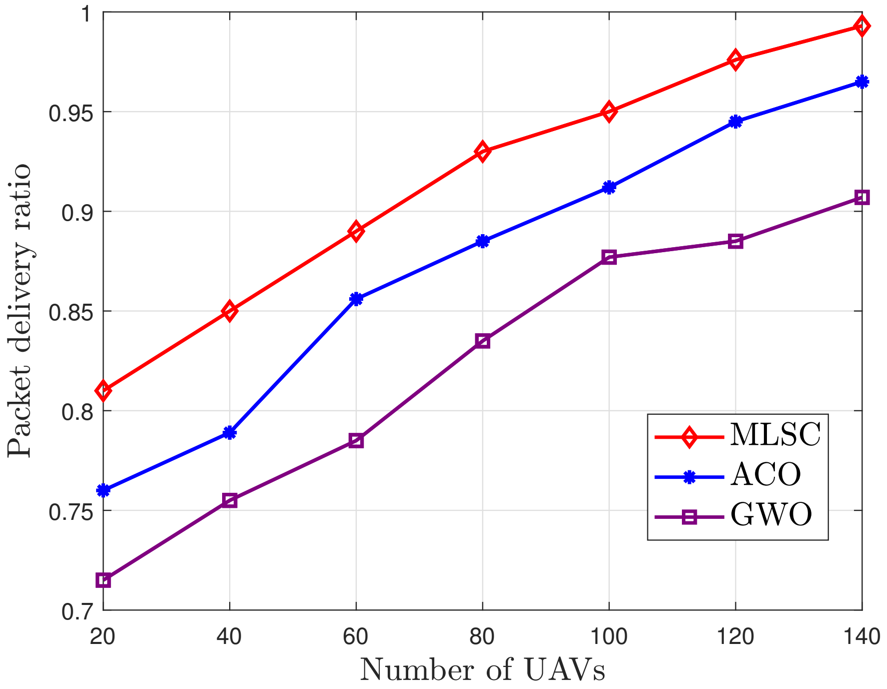

Figure 5.8: Performance comparison of the MLSC scheme with conventional ACO and GWO interms of the packet delivery ratio versus number of UAVs.

high-speed networks. Besides, the AODV floods a route request (RREQ) messages to find a valid path to transmit the data. The RREQ flooding causes unnecessary network overhead that degrades the overall network performance such as packet delivery ratio and network latency. However, the proposed scheme shows comparatively lower normalized routing overhead due to the distributed network formation, where only CH node is involved in the route discovery procedure.

5.4.4 Packet Delivery Ratio

Figure 5.8 presents the performance comparison of the proposed MLSC scheme with conventional ACO and GWO algorithms in terms of packet delivery ratio (PDR) by varying the number of UAVs. The PDR is defined as the number of packets successfully received by the destination/sink node to the number of packets generated by the source nodes. From the Fig. 5.8, it is observed that the PDR in all three cases increases with the number of UAVs.

However, due to the optimal CH selection algorithm and stable root selection method in the MLSC scheme, the PDR is relatively higher than the conventional ACO and GWO scheme. The proposed scheme clearly illustrates the effectiveness by delivering more than 95 percent of the generated packets to the sink. This also demonstrates that the proposed scheme effectively selects a stable CH and backup CH to maintain the stable cluster structure as compared to the other algorithms.

5.4.5 End to End Delay

Figure 5.9 shows the end to end delay comparison of the MLSC scheme with conventional methods by varying the number of UAVs. It is observed that the average delay increases with the number of UAVs. Each UAV in the network begins to experience packet drops and congestion problems due to a large number of UAVs. Subsequently, the link connection of routing route disconnects frequently due to the mobility of UAVs. Moreover, due to the employed

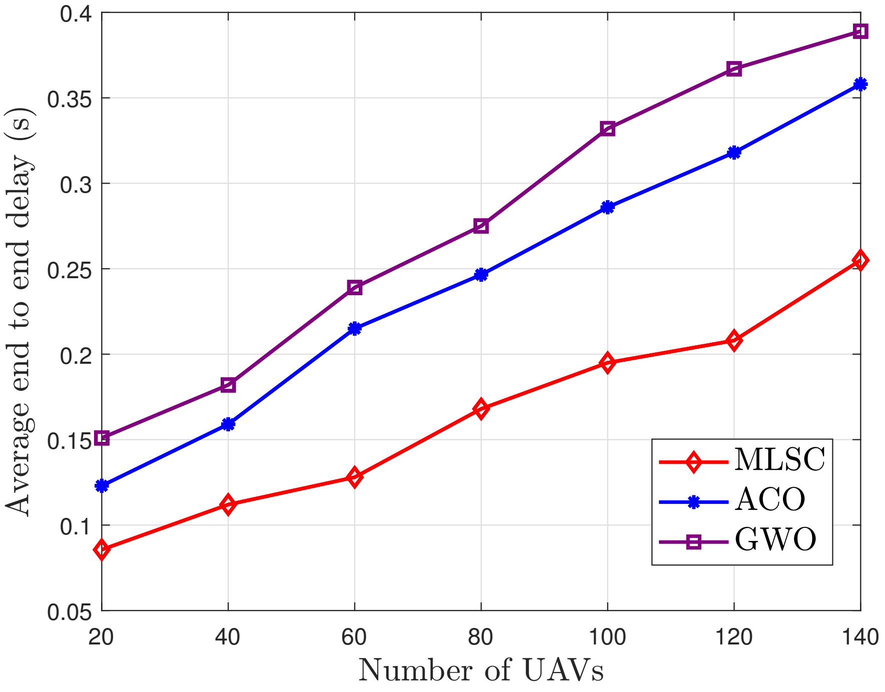

Figure 5.9: Performance comparison of the MLSC scheme with conventional ACO and GWO in terms of average end to end delay versus number of UAVs.

optimal CH and route selection scheme, the average end to end delay of the proposed MLSC scheme is lowest as compared to the conventional ACO and GWO scheme. In the proposed scheme, UAVs transmit the data to their CH, which is located in the optimal position. The packets are collected to CHs by the shortest path routing. Afterwards, the collected packets are forwarded to the sink using a stable backbone tree.

5.5 Chapter Summary

Due to the dynamic topology and high mobility of the UAVs, the conventional protocols which are designed for the stable network are not suitable for UAV networks. The conventional methods will lead to network instability and also increase the network overhead. In this chapter, a location-based distributed clustering algorithm is proposed to enhance the performance and reliability of the UAV networks within resource constraints. The number of UAVs are organized into the clusters. Within the cluster, the data are collected to the CH, and forwarded to the sink/ground station following the backbone tree. First, an analytical model is presented to find the coverage probability of CH and the optimal number of CHs that enhances the network coverage and minimizes the number of transmissions. Then, the clustering algorithm based on the results from the analytical model is proposed. With the help of the simulation results, it has been shown that the proposed scheme substantially improves the network overhead in comparison to the conventional AODV. Moreover, the significant performance is achieved in terms of PDR and average end to end delay as compared to the conventional ACO and GWO schemes.

Chapter 6

Edge-Facilitated Wireless Collaborative Computing in UAV Networks

6.1 Introduction

In recent years, the Unmanned Aerial Vehicles (UAVs) have been widely used in various military and civilian applications, such as surveillance, search and rescue, and communication relaying. The rapid development and onboard integration of low-cost wireless technologies, such as Wi-Fi, IEEE 802.15.4, Global Position System (GPS), and different sensors makes UAV cost-effective for many applications [104]. However, the mobility and dynamics of UAVs causes frequent changes in the network topoogy, which further exacerbate resource constraints of the UAVs. During the data exchange within a UAV networks, one key challenge is to minimize the energy and network latency. The overall performance of the UAV network involves efficient and dynamic resource allocations, such as channel/bandwidth allocation, access selection, and transmission power control. The conventional resource management scheme for wireless networks with Channel State Information (CSI) cannot be directly applied to UAV networks due to the mobility and dynamics of the network. The node failure and link breakages affect the overall communication. The main challenge is to establish and maintain the communication links between UAVs as well as with the ground station due to the mobility [105].

The communication links between UAVs are closely related to the operational environment, such as building height/density and other coexisting networks [106, 107]. Besides, the conventional optimization schemes can not be solved the joint adjustment of UAV's data transmission and flight movement due to the dynamic environment, and also require high computational resources.

The emerging computation-intensive applications (e.g., target detection, automatic navigation, etc.) imposes a great challenge in UAV networks due to the low onboard computation capability and limited battery capability [108]. A Mobile Edge Computing (MEC) has been proposed as a promising technology for new revenue generating 5G use cases since it can improve the computation capacity of computation hungry applications, such as, video surveillance and target detections [109]. The edge computing has been regarded as an effective technology to enhance computing as well as storage capabilities in mobile networks [110, 111]. Moreover, MEC enhances the Quality of Service (QoS) by reducing the congestion on mobile networks and the latency as compared to the conventional Mobile Cloud Computing (MCC). In MCC, devices offload their tasks to the remote cloud servers, and the servers execute the computing task and send back results to the end devices. The MCC provides high resource capacity but fails to provide latency-critical computing services due to the high transmission and propagation delays between the end devices and the cloud server. However, the edge networks can provide efficient computing services to the latency-sensitive application with short propagation and transmission delays as compared to the remote cloud server system.

The overall latency includes the computational delay of the server, queuing delay of the server, and the transmission delay of the packets. The MEC reduces the transmission delay and the queuing delay by placing the computing service locally without transmitting to the remote core network, whereas the computational resource of the server is fixed and determined with the computational tasks [112]. Dividing the entire network into the clusters can effectively limit the channel contention among cluster members to ensure fair channel access. The Cluster Heads (CHs) are responsible for the intra-cluster traffic and performing the task processing at the CH reduces the number of transmissions which minimizes the transmission and queuing delays.

With the MEC, UAVs can offload the tasks to the servers which are located at the edge of the network. Since MEC server can be deployed in the CH, a network with MEC can provide UAVs with low communication latency and save energy [113]. The MEC technology remarkably improves the computing capabilities of mobile devices by offloading the computation-intensive tasks to the MEC server for low-latency computing [114,115]. In addition, collaborative multi-task among the MEC servers can be considered to take full advantage of computational and communication resources [116].

In this Chapter, a heterogeneous set of UAVs sharing a common access point collaborates to perform a set of tasks. Using the distributed computing framework, the tasks are optimally distributed amongst the UAV nodes with the objective of minimizing the total energy consumption of the nodes while satisfying a latency constraint. The main objective is to derive optimal collaborative-computing scheme takes into account both the computing capabilities of the UAV nodes and the strength of their communication links. Owing to the unacceptable delay of MCC, and in the absence of a MEC server nearby UAV networks, the computing and storage capabilities of wireless devices are limited [117]. It might thus be the case, for example, that information is too large to fit in the memory of a single UAV node, or that the UAV nodes are not individually powerful enough to satisfy the latency constraint. To overcome those limitations, a collaborative-computing scheme based on the distributed computing framework [118] is proposed. This distributed computing model involves local computations at the UAV nodes and communication between the nodes via the Access Point (AP)/CH (i.e., the edge of the network is facilitating the communication between the nodes).

The rest of this chapter is organized as follows. Section 6.2 summarizes the related works and provides the literature review. Section 6.3 presents the system model of the collaborative computing in UAV networks. In Section 6.4, proposed scheme and the optimization problem formulation is described in detail. The performance of the proposed scheme is evaluated and compared with the state of art schemes via simulations in Section 6.5. Finally, Section 6.6, concludes the chapter.

6.2 Related Works

Due to the mobility of UAVs, the integration of UAV-enabled communication with MEC can further improve the computation performance [119]. The UAV-enabled MEC architecture was first proposed in [120] which showed that the computation performance can be improved with UAVs. Jointly optimizing bit allocation and UAV's trajectory, the authors in [121] minimized the total mobile energy consumption while satisfying QoS requirements of the offloaded mobile application. Considering wireless power transfer, the computation rate maximization problem was studied in [119] for a UAV-enabled MEC wireless powered system, subject to the energy harvesting causal constraint and the UAV's speed constraint. Moreover, the information exchanges among UAVs increases the communication data traffic and are with the differential QoS requirement [122].

The prior works on wireless distributed computing are mainly focus on the trade-off between the computation and communication loads incurred by the collaboration [123–126]. However, in this chapter, we deal with the set of UAV nodes to be heterogeneous in terms of computing capabilities and communication links, and also added an explicit latency constraint. Motivated by the fact that wireless devices are often limited in energy and that most computing tasks are accompanied by a latency constraint, this work shifts focus towards optimizing the collaborative-computing scheme to minimize the total energy consumption of the UAV nodes, while satisfying the QoS requirements, such as latency constraint in the given network conditions. Besides, a low complexity solution is presented to determine the optimal placement of multiple CH UAVs in a scenario of network capacity enhancement, such as an event happening in an urban area. The *k*-means clustering scheme is utilized to find the optimal placement of multiple CH UAVs. Besides, a new graph embedding based wireless link scheduling mechanism is proposed to find the shortest distance the CHs to transfer the information to deal with the wireless link scheduling problem.

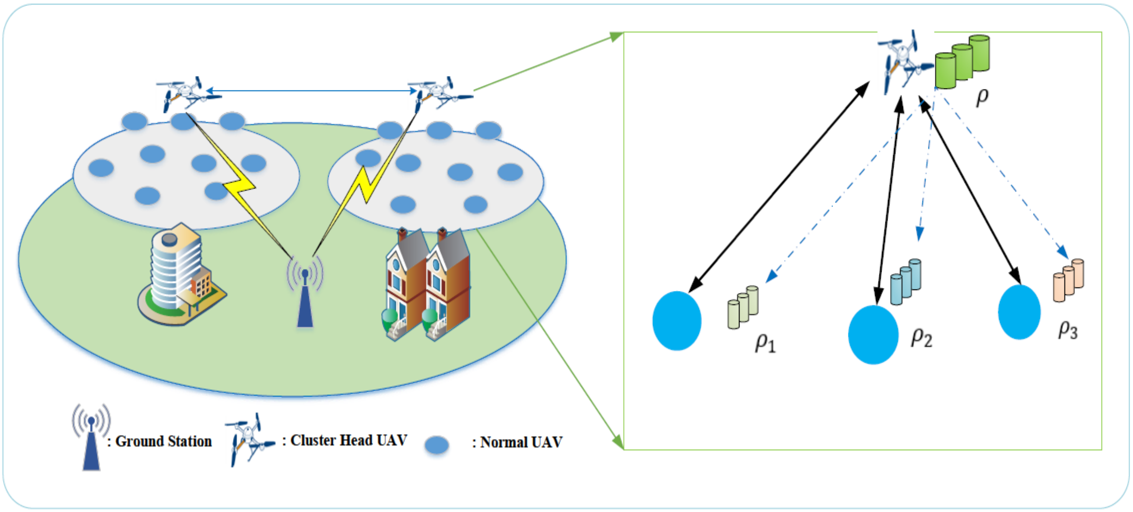

Figure 6.1: Illustration of the clustering and distributed computing model.The UAVs are first divided into different clusters. The CH of each cluster is responsible for inter as well as intra cluster communication. Each UAV node k computes intermediate values $\varrho_k(\delta_l,\ \rho_k)\}_{k=1}^{K}$ and transmitted to the CH. The CH combines the intermediate values to obtain $\varsigma(\delta_l, \rho)$.

6.3 System Model

The proposed distribution model is shown in Fig. 6.1. A set of N UAV nodes, indexed by the letter $n \in [N]$, transmit the information to a common Base Station (BS)/Ground Station (GS). A UAV node can be any device able to wirelessly communicate with the BS and perform local computations. To enhance the stability and accuracy of the network by reducing unnecessary overheads and network latency, we divide the network into the several clusters of size K. The GS has the full knowledge of the network topology and responsible for formulations of the clusters, elections of the CH for each cluster, and constructions of the backbone routing tree. A set of K UAV nodes, idexed by the letter $k \in [K]$ are managed by the corresponding CH. Under a given latency constraint τ , each UAV node k wants to compute a certain function $\chi(\delta_k, \rho)$ where $\delta_k \in [0,1]^D$ the D-bit local information available to UAV node k (e.g., sensed information or local state) and $\rho \in [0,1]^L$ is a L-bit file with $L \gg D$ bits (e.g., a dataset) that might, for instance, be cached at the CH. In the context of environment monitoring, ρ

could be the result of the aggregation over space and time of information sensed from the environment through a network of sensors (e.g., traffic density or temperature) whereas the nodes could be actuators having some local state δ_k that periodically need to perform some latency-sensitive computations to decide whether to take some actions. Other applications include fog computing, mobile crowd-sensing or wireless distributed systems.

The tasks are shared between the K UAV, first, the file ρ can be arbitrarily divided in K smaller files ρ_k (one for each UAV node) of size l_k bits. During the first phase of the framework, each UAV node k computes intermediate values as

$$\varsigma_{k,l} = \varrho_k(\delta_l, \rho_k), \quad l \in [K] \tag{6.1}$$

where $\varrho_k : [0,1]^D \times [0,1]^{l_k} \to [0,1]^{(l_k/L)T}$ is the function executed at node k. The size (in bits) of the intermediate values produced at node k is assumed to be proportional to l_k. Each node k_k thus computes intermediate values for all the other nodes (i.e., $\varsigma_{k,l}$ for all $l \neq k$) and for itself (i.e., $\varsigma_{k,k}$) using the part ρ_k of ρ received from the CH.

Next, the nodes exchange intermediate values with CH. In this phase, each UAV node k transmits the intermediate values $\varsigma_{k,l} = \varrho_k(d_l, \rho_k)$ to node l via the CH, for all $l \neq k$. The UAV node k thus needs to transmit $(K-1)(l_k/L)T$ bits of intermediate values to the CH.

In final phase, each CH l combines the T bits of intermediate values $\{\varsigma_{k,l} = \varrho_k(d_l, \rho_k)\}_{k=1}^{K}$ as

$$\chi(\delta_l, \rho) = h(\varrho_1(\delta_l, \rho_1), \varrho_2(\delta_l, \rho_2), \ldots, \varrho_K(\delta_l, \rho_K)) \tag{6.2}$$

6.4 Proposed Scheme and Problem Formulation

The proposed framework is presented in Fig. 6.2. At first, the UAV network is divided into different clusters using the k-means clustering method and optimal locations of CHs are determine based on the altitude and antenna directivity. Secondly, to reduce overall communication latency we present the graph-based link scheduling algorithm. Finally, to reduce the overall task execution time and energy the optimum collaboratively scheme is proposed. The details

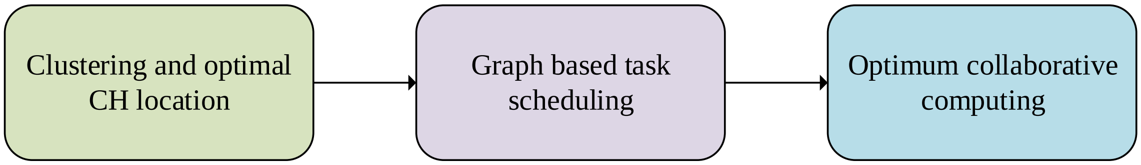
Clustering and optimal CH location
Graph based task scheduling
Optimum collaborative computing

Figure 6.2: The overall proposed framework of the collaborative computing scheme.

of each task are presented in the following subsections.

6.4.1 Clustering and Optimal CH Location

In this section, the optimal position of the CH UAVs is calculated to enhance the capacity of the network in a dense environment. First we have to find the number of CH UAV N_{CH} requried to serve the overall UAV network.

$$N_{CH} = \frac{T_D - T_E}{T_{UAV}}, \tag{6.3}$$

where T_D is the demanded capacity by the users, T_E is the existing network capacity, and T_{UAV} is the throughput of the each UAV, respectively. To find the best location of the CH UAV, a k-means clustering algorithm is used to determine the cluster center. The CH UAVs are positioned at the cluster centers and their altitude is determined to optimize the QoS metrics. From the Fig. 6.3, the coverage radius R can be determined as a function of altitude and the antenna directivity [127], and given by;

$$R = h_{\mathrm{d}} \tan\left(\frac{\theta}{2}\right). \tag{6.4}$$

The flight alittude of the CH UAV is obtained as the function of α and $\alpha \in [0, 1]$ and can be tuned according to the QOS performance, such that

$$h_{\mathrm{d}} = 2\alpha R_{\mathrm{x}} / \tan\left(\frac{\theta}{2}\right), \tag{6.5}$$

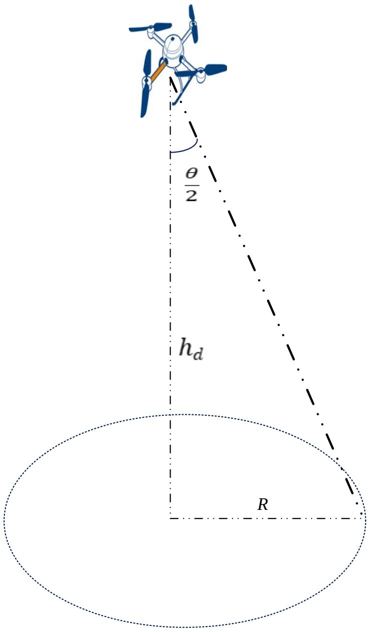

Figure 6.3: CH UAV coverage calculation scenario.The coverage is determined as a function of altitude and the antenna directivity.

where R_x is a radius which is depends on the adopted strategy.

6.4.2 Graph based Link Scheduling

By exploring the graph representation method for the network, the UAV network is modeled as a weighted graph $G(V, E, \alpha)$, which is composed of a set of UAV nodes, V, and a set of edges, E. The edge $e(u, v) \in E$ connects two nodes, $u, v \in V$, and has corresponding weight $\alpha(u, v)$. If each edge has a direction, the graph is referred to as a directed graph. The node features and edge weights depends on the channel gains or the distances between the two nodes of the corresponding communication [128,129]. A set S of CHs is obtained from the above clustering scheme, and we introduced a graph $G_{CH} = \langle U_{CH}, D_{CH} \rangle$, where U_{CH} consists of the set S of CHs. The distance D_{CH} is the shortest path between (CH_i, CH_j) in G. Then, we calculate the Minimum Spanning Tree (MST) of the G_{CH}, and formulate the routing tree between all CHs. In the auxiliary graph $G_{CH} = \langle U_{CH}, D_{CH} \rangle$, and each CH UAV in has an edge e to the each of the UAVs in its neighboring cluster [104]. The distance of each edge (u, v) in E is taken in non decreasing order such that the total distance from the member UAVs in A to their nearest CH UAV is minimized, which reduces overall communication latency. The backbone tree construction mechanism is presented in Algorithm 6.

6.4.3 Problem Formulation

The each UAV have to perform some local computations. If the total number of CPU cycles required to process 1-bit of input data is ζ_k and the energy consumed per CPU cylce is φ_k, then the energy consumed at UAV during the task calculation and the task combined at CH UAV is given by (6.6) and (6.7), respectively.

$$E_{k,UAV} = (KD + l_k)\zeta_k\varphi_k \tag{6.6}$$

$$E_{k,CH} = T\zeta_k\varphi_k \tag{6.7}$$

Similarly, ψ_k be the number of CPU cycles per second, the amounts of time required for the task calculation and combination is given by (6.8) and (6.9), respectively.

Algorithm 6 Construction of the backbone tree for data transmission to the CHs

```
Input: Set $S$ of CHs, and distance $D_{CH}$ between CHs
Output: List of edges $A$ in MST to connect all CHs
Initialize: $G_{CH} = \langle U_{CH}, D_{CH} \rangle$
A ← ∅
for each vertex $v \in V[U_{CH}]$ do
    MAKE-SET($v$)
Sort the edge nodes of $D_{CH}$ into non decreasing order by locations
for each edge $(u, v) \in E$, taken in non decreasing order by minimum distance do
    if FIND-SET($u$) ≠ FIND-SET($v$) then
        A← $A \cup (u, v)$
        UNION $(u, v)$
Return A
```

$$t_{k,UAV} = (KD + l_k)\zeta_k/\psi_k \tag{6.8}$$

$$t_{k,CH} = T\zeta_k/\psi_k \tag{6.9}$$

Let $t_{k,Tm}$ be the time required by UAV to transmit the $(K-1)(l_k/L)T$ bits of intermediate values to CH , the energy consumed at UAV to transmit the intermediated values is calculated as [130]

$$E_{k,Tm} = \gamma_k t_{k,Tm} = \frac{t_{k,Tm}}{|h_k|^2} f\left(\frac{\alpha l_k}{t_{k,Tm}}\right), \tag{6.10}$$

where γ_k and h_k denote the transmit power and wireless channel of node k. The total delay experience by the UAV u for the task completion is given by

$$t_u = t_{k,UAV} + t_{k,UAV} + t_{k,T_m} \tag{6.11}$$

Similarly, the energy consumption is calculated as

$$E_u = E_{k,UAV} + E_{k,Tm} + E_{k,CH} \tag{6.12}$$

In the cloud-edge mobile computing system, the QoS features are characterized by the task completion time and the total energy consumption [131]. The system performance improvement is depending on the QoS features; therefore the utility function can be define as

$$\Omega = \left(\gamma_u^t \frac{t_u^l - t_u}{t_u^l} + \gamma_u^e \frac{E_u^l - E_u}{E_u^l}\right) \forall u \in \mathcal{U}, \tag{6.13}$$

where γ_t^u, $\gamma_u^e \in [0, 1]$, with $\gamma_t^u + \gamma_u^e = 1$, $\forall u \in \mathcal{U}$, specify UAV u's preference on task completion time and energy consumption, respectively. The main objective is to optimize the collaborative-computing scheme to minimize the total energy consumption, while satisfying the latency constraint τ, and can be formualted as [132]

$$\underset{\{l_k\},\{t_{k,Tm}\}}{\text{minimize}} \sum\nolimits_{k=1}^{K} E_{k,UAV} + E_{k,Tm} + E_{k,CH}$$

$$\begin{aligned} \text{subject to } & l_k, t_{k,Tm} \geq 0,\ k \in [K] \\ & t_{k,UAV} + t_{k,Tm} \leq \tau - \max_k\{t_{k,CH}\},\ k \in [K] && (6.14) \\ & \sum\nolimits_{k=1}^{K} l_k = L. && (6.15) \end{aligned}$$

By substituting equations (6.6)-(6.10) in optimization problem and removing constant term, we obtain

$$\underset{\{l_k\},\{t_{k,Tm}\}}{\text{minimize}} \qquad \sum\nolimits_{k=1}^{K} l_k \zeta_k \varphi_k + \frac{t_{k,Tm}}{|h_k|^2} f(\frac{\alpha l_k}{t_{k,Tm}}) \tag{6.16}$$

$$\begin{aligned} \text{subject to} \qquad & l_k, t_{k,Tm} \geq 0, \quad k \in [K] \\ & l_k \frac{\zeta_k}{\psi_k} + t_{k,Tm} \leq \tau_k, \quad k \in [K] && (6.17) \\ & \sum\nolimits_{k=1}^{K} l_k = L. \end{aligned}$$

The objective function (6.16) is always decreasing with $t_{k,Tm}$. Also, a fixed number of bits αl_k to transmit during transmission phase, increasing the duration of $t_{k,Tm}$. The optimization variables can further reduce by substituting l_k by $\frac{\psi_k}{\zeta_k}(\tau_k - t_{k,Tm})$, and given as

$$
\begin{aligned}
\underset{\{t_{k,Tm}\}}{\text{minimize}} \quad & \sum\nolimits_{k=1}^{K} (\tau_k - t_{k,Tm})\psi_k\varphi_k \\
& + \frac{t_{k,Tm}}{|h_k|^2} f\left(\alpha \frac{\psi_k}{\zeta_k}\left(\frac{\tau_k}{t_{k,Tm}} - 1\right)\right) \\
\text{subject to} \quad & 0 \leq t_{k,Tm} \leq \tau_k, \quad k \in [K]
\end{aligned}
$$

$$
\sum\nolimits_{k=1}^{K} \frac{\psi_k}{\zeta_k}(\tau_k - t_{k,Tm}) = L. \tag{6.18}
$$

The above optimization problem can be solved by using the partial Lagrangian and Binary search method [133]. The partial Lagrangian can be define as

$$
\begin{aligned}
\mathcal{L}(\{t_k\}, \lambda) = \sum\nolimits_{k=1}^{K} (\tau_k - t_k)\psi_k\varphi_k + \frac{t_k}{|h_k|^2} f\left(\alpha \frac{\psi_k}{\zeta_k}\left(\frac{\tau_k}{\zeta_k}\right)\right) \\
+ \lambda\left(L - \sum_{k=1}^{K} \frac{\psi}{\zeta}(\tau_k - t_k)\right),
\end{aligned} \tag{6.19}
$$

where $t_{k,Tm}$ has been replaced by t_k to ease notations and with α the Lagrange multiplier associated to (6.18). Then, applying the KKT conditions to the partial Lagrangian leads to

$$
\begin{aligned}
\left.\frac{\partial \mathcal{L}}{\partial t_k}\right|_{*} &= -\psi_k\varphi_k + \frac{1}{|h_k|^2} f\left(\alpha \frac{\psi_k}{\zeta_k}\left(\frac{\tau_k}{t_k^*} - 1\right)\right) \\
&\quad - \frac{\alpha}{|h_k|^2}\frac{\psi_k}{\zeta_k}\frac{\tau_k}{t_k^*} f'\left(\alpha \frac{\psi_k}{\zeta_k}\left(\frac{\tau_k}{t_k^*} - 1\right)\right) + \lambda^* \frac{\psi_k}{\zeta_k} \\
&= -\psi_k\varphi_k - \frac{\Gamma\sigma^2}{|h_k|^2} + \lambda^* \frac{\psi_k}{\zeta_k} \\
&\quad + \frac{\Gamma\sigma^2}{|h_k|^2}\left(1 - \alpha \frac{\ln(2)}{B}\frac{\psi_k}{\zeta_k}\right) 2^{\frac{\alpha}{B}\frac{\psi_k}{\zeta_k}\left(\frac{\tau_k}{t_k^*} - 1\right)} \\
&\begin{cases}
> 0, & t_k^* = 0 \\
= 0, & t_k^* \in]0,\ \tau_k] \\
< 0, & t_k^* = \tau_k \Rightarrow l_k^* = 0,
\end{cases}
\end{aligned} \tag{6.20}
$$

Algorithm 7 Binary search algorithm

```
1: $(\varkappa_l, \varkappa_h) = (0, max_k \left\{\zeta_k \varphi_k + \alpha \frac{\Gamma\sigma^2}{|h_k|^2} \frac{\ln(2)}{B}\right\})$;
2: $(L_l, L_h) = (\sum_k \frac{\psi_k}{\zeta_k}(\tau_k - t^*_{k,l}), \sum_k \frac{\psi_k}{\zeta_k}(\tau_k - t^*_{k,h}))$ where $t^*_{k,l}$ and $t^*_{k,h}$ are obtained using (6.19) with
   $\varkappa_l$ and $\varkappa_h$, respectively.;
3: while $L_l \neq L$ and $L_h \neq L$ do;
4:     $L_m = \sum_k \frac{\psi_k}{\zeta_k}(\tau_k - t^*_{k,h})$ is obtained using (6.22);
5:     with $\varkappa_m = (\varkappa_l + \varkappa_h)/2$.;
6:     if $L_m \geq L$ then, $\varkappa_h = \varkappa_m$, compute $L_h$ as in step 2.;
7:     else if $L_m \leq L$ then, $\varkappa_l = \varkappa_m$, compute $L_l$ as in step 2.;
8:     else $\varkappa^* = \varkappa_m$;
9: end while
```

with

$$\sum\nolimits_{k=1}^{K} \frac{\psi_k}{\zeta_k}(\tau_k - t^*_k) = L.$$

and can be written as

$$\zeta_k \psi_k + \alpha \frac{\Gamma\sigma^2}{|h_k|^2} \frac{\ln(2)}{B} > \lambda^*. \tag{6.21}$$

The left-hand side of the inequality corresponds to the marginal energy consumption of node k per bit received, when node k hasn't received any bit yet, i.e., at $l_k = 0$. The first term corresponds to the marginal energy consumption incurred by the local computational phase while the second term corresponds to the marginal energy consumption incurred by the data transfer phase. In other words, the left-hand side of (6.21) can be interpreted as the "price to start collaborating". If this price is greater than a threshold given by $\varkappa^*$, then $l^*_k = 0$. Finally, solving the remaining case (i.e.,=0) for t^*_k leads to

$$t^*_k = \frac{\alpha \frac{\ln(2)}{B} \frac{\psi_k}{\zeta_k} \times \ \tau_k}{W_0 \left\{ \frac{1}{e} \left(\frac{|h_k|^2}{\Gamma\sigma^2} \frac{\psi_k}{\zeta_k} (\lambda^* - \zeta_k \varphi_k) - 1 \right) e^{\alpha \frac{\ln(2)}{B} \frac{\varphi_k}{\zeta_k}} \right\} + 1} \tag{6.22}$$

where $W_o(.)$ is the main branch of the Lambert function [133]. The optimization problem can then be solved using a one-dimensional search for $\alpha*$, as described in Algorithm 7 [131,

Table 6.1: Simulation Parameters

Parameters	Values
C_k	([500, 1500]) CPU cycles/bit
P_k	([10, 200]) pJ/CPU cycle
F_k	([0.1, 0.2, ..., 1.0]) GHz
h_k	(0, 10^{-3}) Rayleigh fading
B	15 KHz
UAV speed	10-30 m/s
Number of ground station	1
Number of UAVs	10-90
Mobility model	Gauss-Markov mobility model

133].

6.5 Simulation Results

In this section, the simulation results of the proposed scheme are analyzed and evaluated. The simulation parameters are presented in Table 6.1. The Fig. 6.4 shows the UAV deployment scenario in the test environment, similarly, the Fig. 6.5 shows the optimal network scenario after clustering. The each CH are placed in the optimal location to optimize the QoS met-

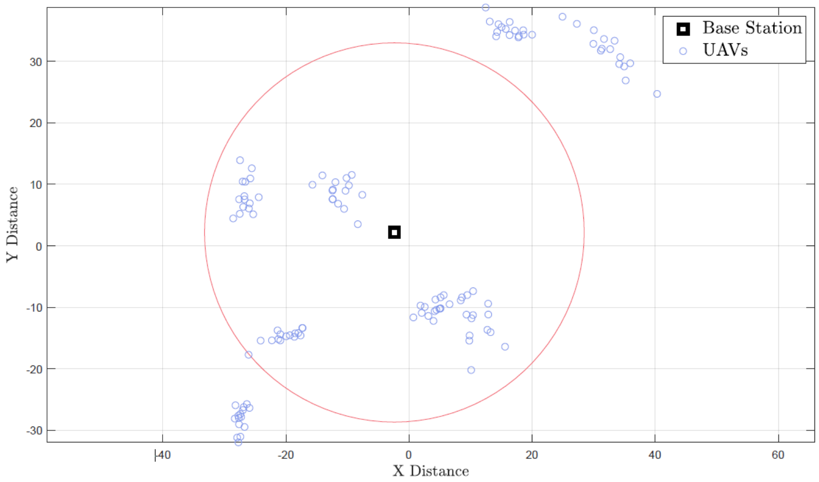

Figure 6.4: Deployment of UAV networks without clusters and CHs.

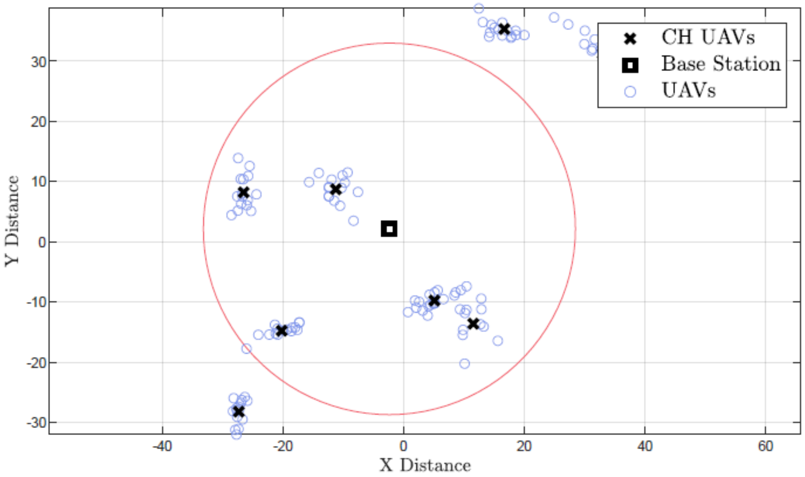

Figure 6.5: Optimal CHs UAV placement based on the coverage radius, altitude and antenna directivity.

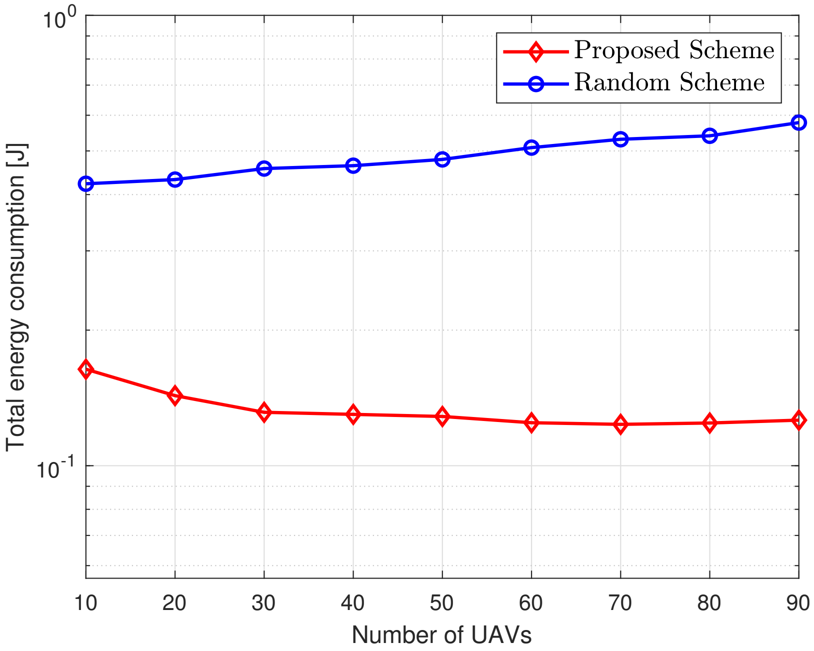

Figure 6.6: Performance comparison of the proposed collaborative computing scheme with random scheme in terms of total energy consumption versus number of UAV nodes.

rics. Fig. 6.6 shows the performance comparision of the proposed scheme and the random resource allocation scheme with $L = 4Mb$, $D = 100b$, and the allowed latency $\tau = 1s$ to ensure feasibility of both schemes. It is observed that the energy consumption is increases with the number of UAVs. However, due to the employed optimal CH location and short route selection scheme, the average energy consumption for resource sharing of the proposed scheme is low as compared to the random scheme.

6.5.1 Enery Consumption versus Number of UAVs

Figure 6.7 presents the energy consumptions of both the proposed and random schemes in different stages of the collaboration. From the figure, it shows that the main energy is consumed during the local computation of individual UAV nodes and task combination phase at the CH. However, the proposed scheme has better performance as compared to the random scheme due to the proposed clustering and optimal placement of the CH scheme. Also, the energy consumptions during the communication increases with the number of UAV nodes. In the beginning, when the number of nodes is less than 40, the random scheme has good perfor-

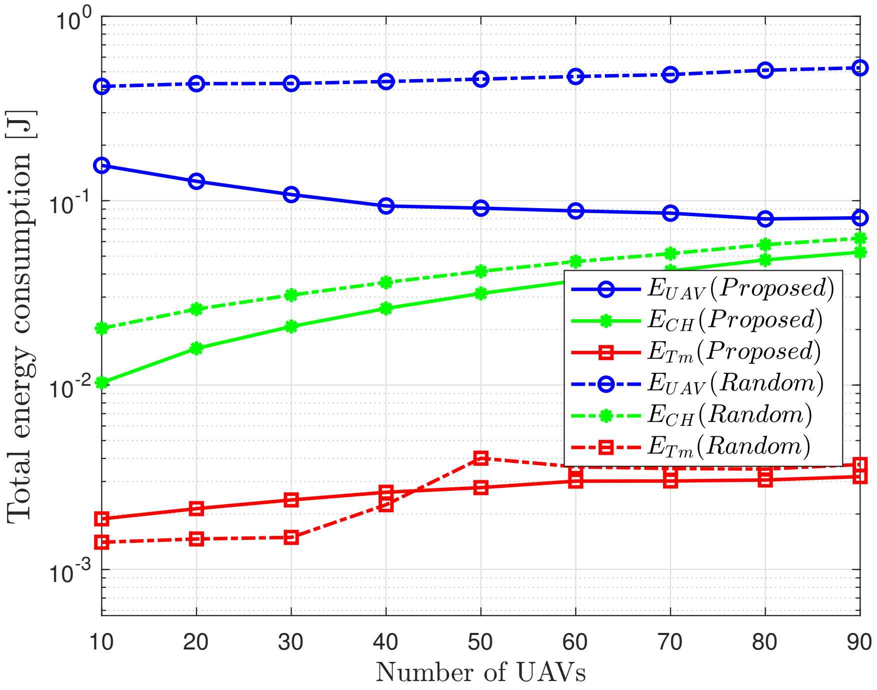

Figure 6.7: Total energy consumed by the local computation of each UAV, communication with CH and task combination at CH for both proposed scheme and random scheme versus Number of UAV nodes.

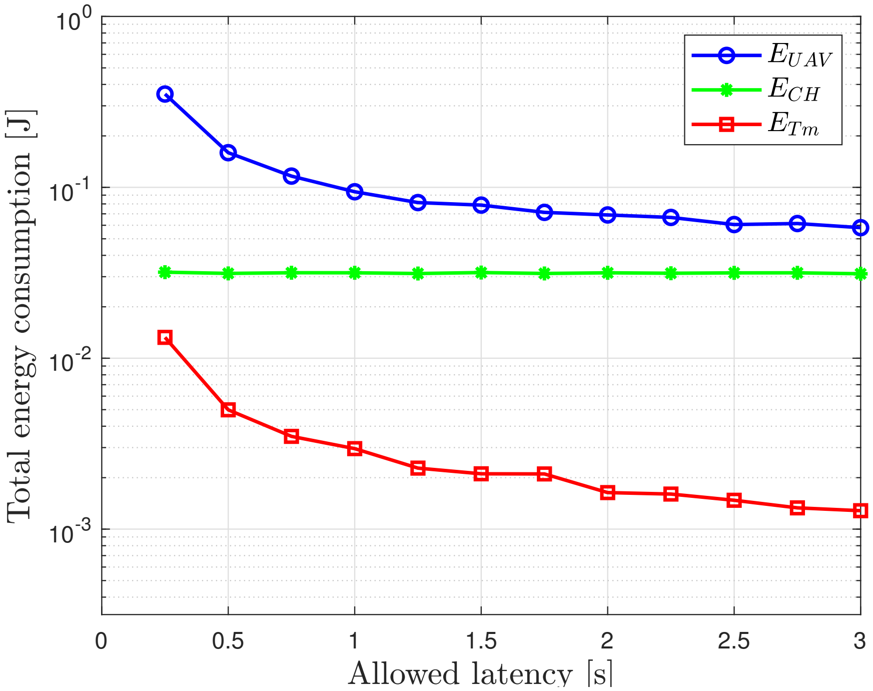

Figure 6.8: Total energy consumed of proposed collaborative scheme as a function of latency for number of nodes =60.

mance than the proposed scheme. However, when the number of UAV nodes is greater than 40, the proposed scheme performed better than the random scheme because of the proposed graph-based wireless link scheduling method to find the shortest and best route to transfer the information to CH.

6.5.2 Enery Consumption versus Latency

Fig. 6.8 depicts how the different energy components of the proposed scheme vary with the latency constraints τ. The proposed scheme able to subside the energy consumption during the transmission phase with the increasing τ. As a result, it concludes that increasing the latency constraints allows the proposed scheme more energy-efficient, and hence the energy consumption decreases.

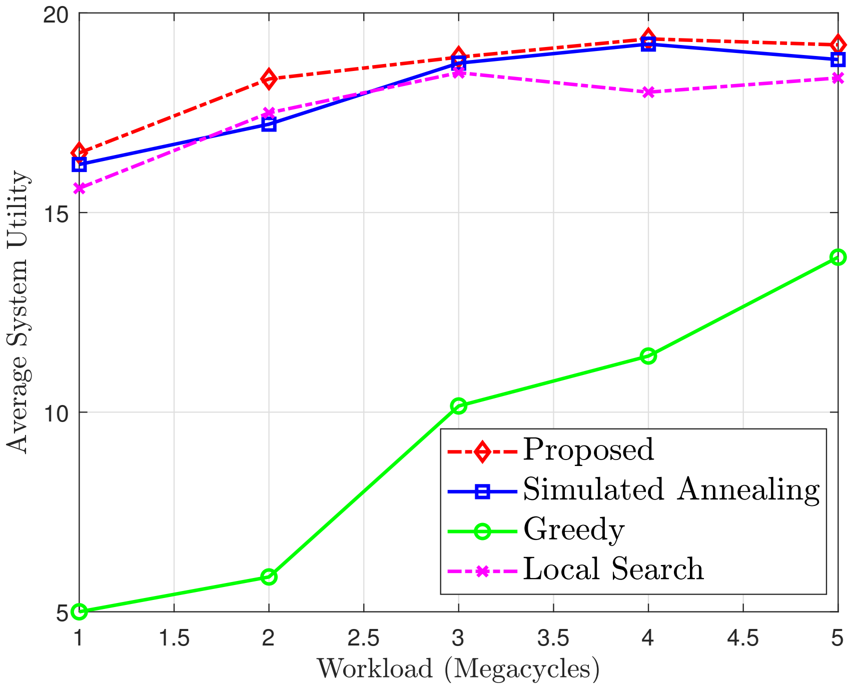

Figure 6.9: Performance comparison of the proposed collaborative computing scheme with state of the art algorithms in terms of average system utility versus workloads (Megacycles).

6.5.3 System Utility versus Workloads

The system utility performance versus offloaded tasks/workloads is presented in Fig. 6.9. The preference parameters as $\gamma_u^t = 0.2$ and $\gamma_u^e = 0.8$ are used. From the figure, it is observed that the average system utility increases as the workload increases. This implies that the task with a high workload will benefit more than that of low workloads. More importantly, the proposed collaborative scheme always performs best as compared to the other state of art schemes when the workload increases. The UAVs will be more benefited from offloading their tasks to the CH when the tasks required more computational resources.

In Fig. 6.10, the system utility performance with respect to the computation tasks in terms of the input size of the workload is evaluated. From the figure, it is shown that the average system utility decreases with the task input sizes. The system utility of the proposed scheme is always higher than the other mechanisms, however, follows a similar trend. From Figs.6.9 and 6.10, it is noticed that the system utility increases with the task workload, however, decreases

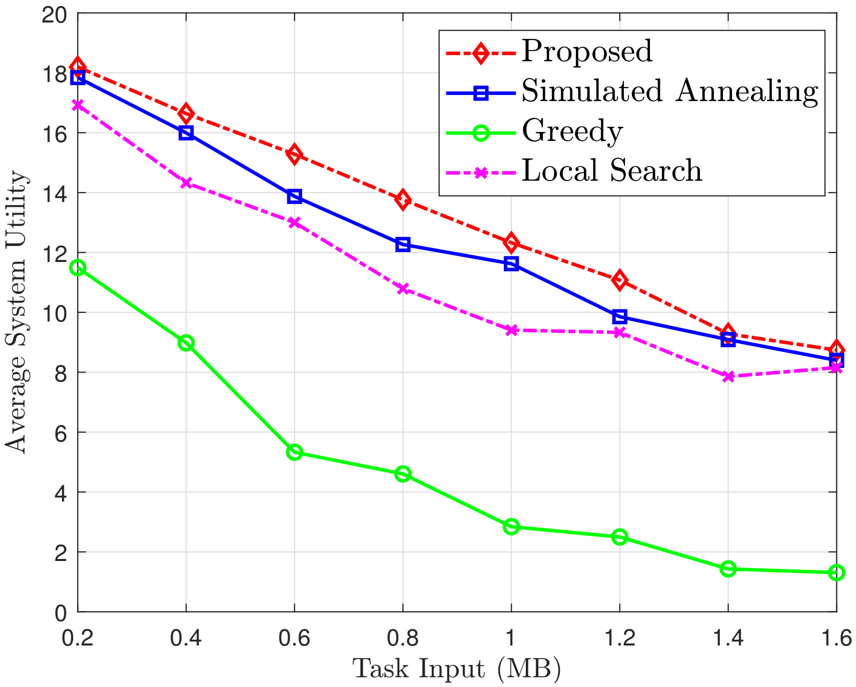

Figure 6.10: Performance comparison of the proposed collaborative computing scheme with state of the art algorithms in terms of average system utility versus task input (MB).

with the task input size. In conclusion, we can say that the tasks with a high workload and small input sizes benefited more than those with low workloads and large input sizes.

6.6 Chapter Summary

Future UAV networks are expected to support a massive number of stations/sensors in diverse applications with different QoS requirements. However, the heterogeneous UAV devices operating in ultra-dense network scenarios may be affected by latency and energy constraints. In this chapter, the low complexity scheme is proposed to determine the optimal location and altitude of CH UAVs to enhance the network capacity by utilizing the *k*-means clustering algorithm. Besides, the graph-based wireless link scheduling algorithm is proposed to find the shortest distance to transfer the information among UAVs to deal with a link scheduling problem. At last, the CH facilitated optimal collaborative computing scheme is derived by considering both the computing capabilities of the UAV nodes and the communication link status. With the help of the simulation results, it has been shown that the proposed edge-facilitated collaborative computing scheme substantially improves overall energy efficiency and latency constraints.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

Future Internet of Things (IoT) networks are expected to support a massive number of heterogeneous devices/sensors in diverse applications ranging from eHealthcare to industrial control systems. In highly-dense deployment scenarios such as Industrial IoT (IIoT) systems and Unmanned Aerial Vehicle (UAV) networks, providing reliable communication links with low-latency becomes challenging due to the involved system delay including data acquisition and processing latencies at the edge-side of IoT networks. In addition, the heterogeneous IoT devices operation in ultra-dense network scenarios may be affected by the packet collisions, delays and dynamic network conditions. In this regard, this thesis investigated the latency minimization, grouping, synchronization, clustering and resource allocation mechanisms in heterogeneous wireless IoT networks to support diverse Quality of Service (QoS) requirements.

The primary contributions of this thesis and their corresponding conclusions are summarized as follows:

In Chapter 2, a cloud-center assisted latency minimization scheme is proposed by using prioritized channel access and data aggregation mechanism. In the proposed scheme, the joint impact of packet scheduling and aggregation is considered by using the preemptive M/G/1 queuing model. A prioritized channel access mechanism is developed by assigning different

Medium Access Control (MAC) layer attributes to the packets based on the applications. In addition, a preemptive M/G/1 queuing model is employed by using separate low-priority and high-priority queues before sending aggregated data to the server. With the help of numerical results, it has been shown that the prioritized channel access and data aggregation scheme provides substantial improvements in terms of latency as compared to the non-prioritized scheme.

In Chapter 3, the sector-based device grouping scheme is proposed for fast and efficient channel access in IEEE 802.11ah based IoT networks. In the proposed framework, the Access Point (AP) divides its coverage area into different sectors, and then each sector is further divided into distinct groups based on a number of the devices and their location information. The individual groups within a sector are assigned to specific Random Access Window (RAW) slots. The sector-based grouping allows the substantial improvement on packet collision rate/probability and throughput of ultra-dense IoT networks by utilizing the spatial orthogonal access mechanism. The simulation results have indicated that the proposed scheme significantly improve the system delay and network throughput as compared to the conventional Distributed Coordination Function (DCF) and IEEE 802.11ah grouping mechanism.

In Chapter 4, an efficient clock synchronization scheme is proposed for the event critical applications in wireless IoT. A priority-based fast and efficient channel access scheme proposed in Chapter 2 is utilized to enhance the synchronization precision of the event critical IoT nodes. The proposed scheme assigns time slots with high preference to the timestamp packets of critical nodes and also guarantees the channel access in event-based situations. Furthermore, the proposed scheme provides the deterministic packet scheduling and reduces the channel access delay as presented in Chapter 2 and enhances the synchronization precision. With the help of the simulation results, it has been shown that the proposed scheme substantially improves the synchronization precision of the event critical sensor nodes as compared to the normal nodes.

The technical challenges in mobile IoT networks such as UAV networks are optimal deployment of UAVs, energy limitations, path planning, interference management, and stable wireless links. The UAV mobility causes the significant impact on the link connectivity of UAV swarm networks. The effective management of UAV swarms also relies on low latency communications. However, due to the frequent topology changes, high mobility, and unstable wireless links make the conventional protocols unreliable in UAV networks. To address the above-mentioned issues, in Chapter 5, a mobility and location-aware stable clustering scheme is proposed for randomly deployed UAVs network by incorporating the mobility and coverage probability. The cluster maintenance mechanism is also presented with reference to the relative mobility and locations to enhance the stability of the cluster network. In addition, the graph-based link scheduling algorithm is proposed to find the shortest distance to transfer the information among the UAVs to overcome the latency and link schedule problems. The performance evaluation has been conducted and the results indicated that the proposed scheme improved the Packed Delivery Ratio (PDR) and end to end delay as compared to the conventional schemes.

Based on the system model and graph-based link scheduling algorithm presented in Chapter 5, the distributed computing framework is proposed in Chapter 6. The tasks are optimally distributed amongst the UAV nodes with the objective of minimizing the total energy consumption of the nodes while satisfying a latency constraint. The main objective is to derive an optimal collaborative-computing scheme that considers both the computing capabilities of the UAV nodes and the strength of their communication links. In addition, the graph-based link scheduling algorithm is used to find the shortest distance to transfer the information among the UAVs to overcome the latency and link schedule problems. The simulation results illustrated that the proposed collaborative computing scheme substantially enhances the system performance by improving latency and energy efficiency.

7.2 Future Work

The technical issues on the ultra-dense IoT system have been addressed in this thesis, where several QoS provisioning mechanisms have been adopted to improve the QoS performance. There are still several challenges that have to be addressed and investigated to improve the QoS performance as well as the overall IoT system according to the application needs. The research problems presented in this thesis can be further extended in several aspects. In this section, some of the novel future research directions including secure grouping mechanism, protocol designs, adaptive QoS provisioning schemes using machine learning methods, and secure channel access mechanism using synchronization are identified and summarized as follows;

7.2.1 Design Issues for Low Power IoT Devices Transmission Schemes

The current protocol designs for low-power IoT devices are based on the assumption of latency requirements. The major problems involved in providing wireless connectivity to low-power IoT devices are battery capacity, transmission range, system capacity, and price. A significant compromise is made in the link performance due to the low cost and low capacity of devices, which affects overall coverage. The coverage loss can be compensated by utilizing the extended transmission interval, however, this method will increase the battery consumption. Therefore, it is very crucial to balance the trade-off between coverage enhancement and battery consumption while designing transmission methods for IoT devices. Moreover, within the given bandwidth, the transmission protocol should be able to handle a large number of devices while ensuring battery efficiency. The effective bandwidth method can be utilized to achieve the balance between spectral efficiency for the given coverage and transmission time. In addition, to support the massive number of devices in the upcoming 5G and beyond technology, advanced transmission scheduling schemes with low signaling overhead Medium Access Control (MAC) protocols need to be investigated.

7.2.2 Intelligence Access Control and Privacy Protection

Several technical issues arise in designing the sensing, learning, and decision-making procedure in IoT systems, including optimization, resource allocation, computational and communication capability, privacy, and data analytics. The Artificial Intelligence (AI) is one of the promising solutions to handle these issues in real-time. The Machine Learning (ML) techniques detect the malicious activity based on the training data and identify the intrusions in real-time by improving the detection efficiency compared to the state of art methods. Several devices are connected to the one Access Point (AP)/gateway in the ultra-dense IoT network, and providing fair access is crucial. The authenticated devices can only access the other devices and data within their authorities and cannot perform other tasks beyond their access authorities. To manage the device level authorities, the potential solution will be an intelligent classification scheme at the edge level. The intelligence scheme will classify the devices in low-privileged or high-privileged devices according to the device features and will help to control the potential attacks. Moreover, decentralized and edge-cloud collaborative schemes need to be further investigated to manage the access control of the ultra-dense heterogenous IoT networks.

7.2.3 Secure Time Synchronization Protocol

Synchronization is crucial for wireless IoT networks, where devices rely on a common time reference for event detection and industrial process control. Wireless communications are vulnerable to attacks such as manipulation, congestions, and eavesdropping. In particular, attacks to the time synchronization service may incur data distortion as well as malfunction the whole network. The design of secure time synchronization protocols still faces nontrivial challenges such as; *i*: resource constraints of devices to perform complex computation, and *ii*: traditional time-synchronization protocols cannot be easily extended to adopt security mechanism since they only rely on a few reference nodes, which often implies susceptibility to single points of failure. To address these challenges many secure time synchronization protocols, identify possible attacks with a heuristic threshold of clock offset, however, complex authentication and encryption operations typically require more computational resources. Due to the limited computing and communication resources of the devices, supporting security mechanisms at the device level is challenging, and can be considered as a potential future topic.

7.2.4 Situation Aware Channel Quality and Wireless Link Stability Estimation

In IoT networks, a single message may reach many receivers due to the broadcast nature of the radio transmissions. The success of message arrival depends on the distance between the transmitter and potential receivers and is also affected by multipath effects, signal attenuation, and signal interference from other wireless communication protocols. The uncertain temporal and spatial characteristics of data transmission present challenges for the prediction and evaluation of wireless link stability. The Link Quality Prediction (LQP) plays a fundamental role in IoT routing protocols, topology control, and energy management. The effective scheme for the link quality prediction model can help to choose the better link for data transmission and improve the network throughput and the reliability of data transmission. The topology control mechanism in IoT relies on link quality to eliminate unnecessary links and improve the stability of the network. The ML approach can be used to predict the temporal quality of the wireless link based on the PHY layer parameters and Packet Received Rate (PRR). The above-mentioned combined parameters resemble the current state of the channel so that the learning model can perform an accurate estimation. The prediction model can be developed by utilizing the ML algorithms related to pattern recognition, linear regression, and support vector machines.

7.2.5 Self Aware Resource Allocation Scheme

Due to the dynamic topology and mobility of UAV, the wireless channel conditions change rapidly over time. The traditional centralized resource allocation scheme for device-to-device (D2D) communications with the assumption of detailed channel state information (CSI) can not be applied in UAV networks since it is impossible to track the channel variations in a short time frame. Moreover, the centralized control mechanisms will incur network overheads and latency issues to get the network-wide information. The transmission overheads and latency increase with the number of UAVs and their mobility. One of the potential solutions is to develop a decentralized intelligent resource allocation framework for UAV networks based on deep reinforcement learning. Each UAV makes its own decisions to find the optimal subband frequency and transmission power level. Besides, the global network information is not required to make the decisions, hence the network overhead and latency will be minimized.

7.2.6 Security Enhancement on Edge-Facilitated Collaborative Computing

The edge facilitated collaborative computing further enhances the QoS performance of IoT networks in several aspects. Sharing the information within the IoT network brings opportunities and benefits, whereas at the same time it also suffers from security and privacy issues. If the information from the devices is inspected by the malicious attacker, the participated edge-cloud user's privacy would be threatened. There are still several privacy and security threats that need to be considered to protect and preserve the confidentiality and integrity of data in the process of task offloading. The generic architecture to provide the security functions in edge facilitated computing paradigms is needed to ensure the integrity and confidentiality of IoT devices.

Bibliography

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, Fourthquarter 2015.

[2] S. K. Sharma, T. E. Bogale, S. Chatzinotas, X. Wang, and L. B. Le, "Physical layer aspects of wireless IoT," in *Proc. IEEE ISWCS*, Sept 2016, pp. 304–308.

[3] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, 2013.

[4] M. Zorzi, A. Gluhak, S. Lange, and A. Bassi, "From today's INTRAnet of things to a future INTERnet of things: a wireless- and mobility-related view," *IEEE Wireless Commun.*, vol. 17, no. 6, pp. 44–51, Dec 2010.

[5] J. Lin and *et al*, "A survey on Internet of Things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1125–1142, Oct 2017.

[6] L. D. Xu, W. He, and S. Li, "Internet of Things in industries: A survey," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2233–2243, Nov 2014.

[7] ITU-R, "Minimum requirements related to technical performance for IMT-2020 radio interface(s)," https://www.itu.int/md/R15-SG05-C-0040/en, Feb. 2017, document 5D/TEMP/300(Rev.1).

[8] M. Kamel, W. Hamouda, and A. Youssef, "Ultra-dense networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 4, pp. 2522–2545, Fourthquarter 2016.

[9] T. C. Chang, C. H. Lin, K. C. J. Lin, and W. T. Chen, "Load-balanced sensor grouping for IEEE 802.11ah networks," in *Proc. IEEE GLOBECOM*, Dec 2015, pp. 1–6.

[10] M. Doudou, D. Djenouri, and N. Badache, "Survey on latency issues of asynchronous MAC protocols in delay-sensitive wireless sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 528–550, Second 2013.

[11] J. Wang, C. Jiang, Z. Han, Y. Ren, R. G. Maunder, and L. Hanzo, "Taking drones to the next level: Cooperative distributed unmanned-aerial-vehicular networks for small and mini drones," *IEEE Veh. Technol. Mag.*, vol. 12, no. 3, pp. 73–82, Sep. 2017.

[12] Z. Han, A. L. Swindlehurst, and K. J. R. Liu, “Optimization of manet connectivity via smart deployment/movement of unmanned air vehicles,” *IEEE Veh. Technol. Mag.*, vol. 58, no. 7, pp. 3533–3546, Sep. 2009.

[13] C. Y. Tazibt, M. Bekhti, T. Djamah, N. Achir, and K. Boussetta, “Wireless sensor network clustering for UAV-based data gathering,” in *Proc. Wireless Days*, March 2017, pp. 245–247.

[14] P. Yadav, J. A. McCann, and T. Pereira, “Self-synchronization in duty-cycled internet of things (IoT) applications,” *IEEE Internet Things J*, vol. 4, no. 6, pp. 2058–2069, Dec 2017.

[15] L. Zheng and *et al*, “Performance analysis of group-synchronized DCF for dense IEEE 802.11 networks,” *IEEE Trans. Wireless Commun.*, vol. 13, no. 11, pp. 6180–6192, Nov 2014.

[16] I. Al-Anbagi, M. Erol-Kantarci, and H. T. Mouftah, “Priority- and delay-aware medium access for wireless sensor networks in the smart grid,” *IEEE Syst. J.*, vol. 8, no. 2, pp. 608–618, June 2014.

[17] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey,” *Comput. Net.*, vol. 54, no. 15, pp. 2787–2805, 2010.

[18] M. Jutila, “An adaptive edge router enabling Internet of Things,” *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1061–1069, Dec 2016.

[19] S. Bhandari, S. K. Sharma, and X. Wang, “Cloud-assisted device clustering for lifetime prolongation in wireless IoT networks,” in *Proc. IEEE CCECE*, April 2017, pp. 1–4.

[20] N. Nasser, L. Karim, and T. Taleb, “Dynamic multilevel priority packet scheduling scheme for wireless sensor network,” *IEEE Trans. Wireless Commun.*, vol. 12, no. 4, pp. 1448–1459, April 2013.

[21] M. M. Hassan, H. S. Albakr, and H. Al-Dossari, “A cloud-assisted internet of things framework for pervasive healthcare in smart city environment,” in *Proc. EMASC*. ACM, 2014, pp. 9–13.

[22] S. K. Sharma and X. Wang, “Live data analytics with collaborative edge and cloud processing in wireless IoT networks,” *IEEE Access*, vol. 5, pp. 4621–4635, 2017.

[23] S. Mubeen, P. Nikolaidis, A. Didic, H. Pei-Breivold, K. Sandstrom, and M. Behnam, “Delay mitigation in offloaded cloud controllers in industrial IoT,” *IEEE Access*, vol. 5, pp. 4418–4430, 2017.

[24] IEEE 802.15.4, “Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for Low Rate Wireless Personal Area Networks (WPANs),” *IEEE Std 802.15.4-2006*, pp. 1–320, Sept 2006.

[25] I. Park, D. Kim, and D. Har, “MAC achieving low latency and energy efficiency in hierarchical M2M networks with clustered nodes,” *IEEE Sensors J.*, vol. 15, no. 3, pp. 1657–1661, March 2015.

[26] H. Yan, Y. Zhang, Z. Pang, and L. D. Xu, “Superframe planning and access latency of slotted MAC for industrial WSN in IoT environment,” *IEEE Trans. Ind. Informat*, vol. 10, no. 2, pp. 1242–1251, May 2014.

[27] S. Jardosh and P. Ranjan, “EPCAP: Explicit Prioritized Channel Access Protocol for IEEE 802.15.4 based wireless sensor networks,” in *Proc. IEEE WCNC*, April 2013, pp. 1–6.

[28] N. Kouzayha, M. Jaber, and Z. Dawy, “M2M data aggregation over cellular networks: signaling-delay trade-offs,” in *Proc. IEEE GLOBECOM Wkshps*, Dec 2014, pp. 1155–1160.

[29] G. Rigazzi, N. K. Pratas, P. Popovski, and R. Fantacci, “Aggregation and trunking of M2M traffic via D2D connections,” in *Proc. IEEE ICC*, June 2015, pp. 2973–2978.

[30] S. A. AlQahtani, “Analysis and modelling of power consumption-aware priority-based scheduling for M2M data aggregation over long-term-evolution networks,” *IET Commun.*, vol. 11, pp. 177–184(7), Jan 2017.

[31] A. Koubaa, M. Alves, B. Nefzi, and Y.-Q. Song, “Improving the IEEE 802.15.4 Slotted CSMA/CA MAC for Time-Critical Events in Wireless Sensor Networks,” in *Proc. RTN, ECRTS Wkshps* , July 2006.

[32] Q. Ye, W. Zhuang, L. Li, and P. Vigneron, “Traffic-load-adaptive medium access control for fully connected mobile ad hoc networks,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 11, pp. 9358–9371, Nov 2016.

[33] H. A. Omar, W. Zhuang, A. Abdrabou, and L. Li, “Performance evaluation of VeMAC supporting safety applications in vehicular networks,” *IEEE Trans. Emerg. Topics Comput.*, vol. 1, no. 1, pp. 69–83, June 2013.

[34] T. O. Kim, J. S. Park, H. J. Chong, K. J. Kim, and B. D. Choi, “Performance analysis of IEEE 802.15.4 non-beacon mode with the unslotted CSMA/CA,” *IEEE Commun. Lett.*, vol. 12, no. 4, pp. 238–240, April 2008.

[35] P. Park, P. D. Marco, P. Soldati, C. Fischione, and K. H. Johansson, “A generalized markov chain model for effective analysis of slotted IEEE 802.15.4,” in *Proc. IEEE MASS*, Oct 2009, pp. 130–139.

[36] M. H. Zayani, V. Gauthier, and D. Zeghlache, “A joint model for IEEE 802.15.4 physical and medium access control layers,” in *Proc. IWCMC*, July 2011, pp. 814–819.

[37] T. Adame and *et al*, “IEEE 802.11ah: the WiFi approach for M2M communications,” *IEEE Wireless Commun.*, vol. 21, no. 6, pp. 144–152, Dec 2014.

[38] N. Nawaz and *et al*, “Throughput enhancement of restricted access window for uniform grouping scheme in IEEE 802.11ah,” in *Proc. IEEE ICC*, May 2017, pp. 1–7.

[39] S. Aust, R. V. Prasad, and I. G. M. M. Niemegeers, “Sector-based RTS/CTS access scheme for high density WLAN sensor networks,” in *Proc. IEEE LCNW*, Sept 2014, pp. 697–701.

[40] E. Khorov, A. Krotov, and A. Lyakhov, “Modelling machine type communication in IEEE 802.11ah networks,” in *Proc. IEEE ICCW*, June 2015, pp. 1149–1154.

[41] L. Tian, J. Famaey, and S. Latre, “Evaluation of the IEEE 802.11ah restricted access window mechanism for dense IoT networks,” in *Proc. IEEE WoWMoM*, June 2016, pp. 1–9.

[42] J. O. Seo, C. Nam, S. G. Yoon, and S. Bahk, “Group-based contention in IEEE 802.11ah networks,” in *Proc. IEEE ICTC*, Oct 2014, pp. 709–710.

[43] S.-G. Yoon, J.-O. Seo, and S. Bahk, “Regrouping algorithm to alleviate the hidden node problem in 802.11ah networks,” *Comput. Netw.*, vol. 105, pp. 22–32, 2016.

[44] S. Aust, R. V. Prasad, and I. G. M. M. Niemegeers, “IEEE 802.11ah: Advantages in standards and further challenges for sub 1 GHz Wi-Fi,” in *Proc. IEEE ICC*, June 2012, pp. 6885–6889.

[45] P. Sthapit and J.-Y. Pyun, “Station grouping strategy for minimizing association delay in IEEE 802.11ah,” *IEICE Trans. Commun.*, vol. E100.B, no. 8, pp. 1419–1427, 2017.

[46] S. Bhandari, S. K. Sharma, and X. Wang, “Cloud-assisted device clustering for lifetime prolongation in wireless IoT networks,” in *Proc. IEEE CCECE*, April 2017, pp. 1–4.

[47] ——, “Latency minimization in wireless IoT using prioritized channel access and data aggregation,” in *Proc. IEEE GLOBECOM*, Dec 2017, pp. 1–6.

[48] P. Chatzimisios, A. C. Boucouvalas, and V. Vitsas, “Packet delay analysis of IEEE 802.11 MAC protocol,” *Electron. Lett.*, vol. 39, no. 18, pp. 1358–1359, Sept 2003.

[49] G. Bianchi, “Performance analysis of the IEEE 802.11 distributed coordination function,” *IEEE J. Sel. A. Commun.*, vol. 18, no. 3, pp. 535–547, March 2000.

[50] Y. Kim and *et al*, “Optimal throughput analysis of a super dense wireless network with the renewal access protocol,” in *Proc. IEEE ICCW*, June 2015, pp. 2194–2199.

[51] F. Lamonaca, A. Gasparri, E. Garone, and D. Grimaldi, “Clock synchronization in wireless sensor network with selective convergence rate for event driven measurement applications,” *IEEE Trans. Instrum. Meas.*, vol. 63, no. 9, pp. 2279–2287, Sept 2014.

[52] K. S. Kim, S. Lee, and E. G. Lim, “Energy-efficient time synchronization based on asynchronous source clock frequency recovery and reverse two-way message exchanges in wireless sensor networks,” *IEEE Trans. Commun.*, vol. 65, no. 1, pp. 347–359, Jan 2017.

[53] M. Zayani, V. Gauthier, and D. Zeghlache, “A joint model for IEEE 802.15.4 physical and medium access control layers,” in *Proc. IEEE IWCMC*, July 2011, pp. 814–819.

[54] Q. Wang and J. Jiang, “Comparative examination on architecture and protocol of industrial wireless sensor network standards,” *IEEE Commun. Surv. Tutor.*, vol. 18, no. 3, pp. 2197–2219, thirdquarter 2016.

[55] S. Bhandari, S. K. Sharma, and X. Wang, “Device grouping for fast and efficient channel access in IEEE 802.11ah based IoT networks,” in *Proc. IEEE ICC Wkshps*, May 2018.

[56] S. Petersen and S. Carlsen, “Wirelesshart versus ISA100.11a: The format war hits the factory floor,” *IEEE Ind. Electron. Mag.*, vol. 5, no. 4, pp. 23–34, Dec 2011.

[57] M. K. Maggs, S. G. O’Keefe, and D. V. Thiel, “Consensus clock synchronization for wireless sensor networks,” *IEEE Sens. J*, vol. 12, no. 6, pp. 2269–2277, June 2012.

[58] T. N. Dinh and P. H. Ha, “Advanced GTS scheduling in IEEE 802.15.4 networks for industrial application,” in *Proc. IEEE CCNC*, Jan 2019, pp. 1–4.

[59] H. Farag, A. Mahmood, M. Gidlund, and P. Oesterberg, “PR-CCA MAC: A prioritized random CCA MAC protocol for mission-critical iot applications,” in *Proc. IEEE ICC*, May 2018, pp. 1–6.

[60] M. Maróti, B. Kusy, G. Simon, and A. Lédeczi, “The flooding time synchronization protocol,” in *Proc. SenSys ’04*. New York, NY, USA: ACM, 2004, pp. 39–49.

[61] B. J. Choi, H. Liang, X. Shen, and W. Zhuang, “DCS: Distributed asynchronous clock synchronization in delay tolerant networks,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 3, pp. 491–504, March 2012.

[62] D. Shrestha, Z. Pang, and D. Dzung, “Precise clock synchronization in high performance wireless communication for time sensitive networking,” *IEEE Access*, vol. 6, pp. 8944–8953, 2018.

[63] S. Youn, “A comparison of clock synchronization in wireless sensor networks,” *Int. J. Distrib. Sens. N.*, vol. 9, no. 12, p. 532986, 2013.

[64] D. L. Mills, “Internet time synchronization: the network time protocol,” *IEEE Trans. Commun.*, vol. 39, no. 10, pp. 1482–1493, Oct 1991.

[65] F. Cristian, “Probabilistic clock synchronization,” *Distributed Comput.*, vol. 3, no. 3, pp. 146–158, Sep 1989.

[66] S. Ganeriwal, R. Kumar, and M. B. Srivastava, “Timing-sync protocol for sensor networks,” in *Proc. of SenSys ’03*. New York, NY, USA: ACM, 2003, pp. 138–149.

[67] J.-P. Sheu, W.-K. Hu, and J.-C. Lin, “Ratio-based time synchronization protocol in wireless sensor networks,” *Telecommun Syst*, vol. 39, no. 1, pp. 25–35, Sep 2008.

[68] J. Elson, L. Girod, and D. Estrin, “Fine-grained network time synchronization using reference broadcasts,” *SIGOPS Oper. Syst. Rev.*, vol. 36, no. SI, pp. 147–163, Dec. 2002.

[69] K. l. Noh, E. Serpedin, and K. Qaraqe, “A new approach for time synchronization in wireless sensor networks: Pairwise broadcast synchronization,” *IEEE Trans. Wireless Commun.*, vol. 7, no. 9, pp. 3318–3322, September 2008.

[70] D. Djenouri, “R^4Syn : Relative referenceless receiver/receiver time synchronization in wireless sensor networks,” *IEEE Signal Process. Lett.*, vol. 19, no. 4, pp. 175–178, April 2012.

[71] J. Åkerberg, M. Gidlund, and M. Björkman, “Future research challenges in wireless sensor and actuator networks targeting industrial automation,” in *Proc. IEEE INDIN*, July 2011, pp. 410–415.

[72] H. Farag, M. Gidlund, and P. Österberg, “A delay-bounded MAC protocol for mission- and time-critical applications in industrial wireless sensor networks,” *IEEE Sens. J*, vol. 18, no. 6, pp. 2607–2616, March 2018.

[73] “IEEE standard for a precision clock synchronization protocol for networked measurement and control systems,” *IEEE Std 1588-2008 (Revision of IEEE Std 1588-2002)*, pp. 1–300, July 2008.

[74] S. P. Chepuri, R. T. Rajan, G. Leus, and A. van der Veen, “Joint clock synchronization and ranging: Asymmetrical time-stamping and passive listening,” *IEEE Signal Process. Lett.*, vol. 20, no. 1, pp. 51–54, Jan 2013.

[75] S. C. Park, M. Tummala, and J. C. McEachen, “Method and apparatus for hybrid time synchronization based on broadcast sequencing for wireless ad hoc networks,” June 2017, US Patent 9,693,325.

[76] R. Pagliari and A. Scaglione, “Scalable network synchronization with pulse-coupled oscillators,” *IEEE Trans. Mob. Comput.*, vol. 10, no. 3, pp. 392–405, March 2011.

[77] R. Ke, Z. Li, J. Tang, Z. Pan, and Y. Wang, “Real-time traffic flow parameter estimation from UAV video based on ensemble classifier and optical flow,” *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 1, pp. 54–64, Jan 2019.

[78] M. Y. Arafat and S. Moh, “Localization and clustering based on swarm intelligence in uav networks for emergency communications,” *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8958–8976, Oct 2019.

[79] ——, “A survey on cluster-based routing protocols for unmanned aerial vehicle networks,” *IEEE Access*, vol. 7, pp. 498–516, 2019.

[80] M. A. Uddin, A. Mansour, D. Jeune, M. Ayaz, and E.M.Aggoune, “UAV-assisted dynamic clustering of wireless sensor networks for crop health monitoring,” *Sensors*, vol. 18, no. 2, 2018.

[81] I. Jawhar, N. Mohamed, J. Al-Jaroodi, and S. Zhang, "A framework for using unmanned aerial vehicles for data collection in linear wireless sensor networks," *J. Intell & Robot. Syst.*, vol. 74, no. 1, pp. 437–453, Apr 2014.

[82] F. Aadil, A. Raza, M. F. Khan, M. Maqsood, I. Mehmood, and S. Rho, "Energy aware cluster-based routing in flying ad-hoc networks," *Sensors*, vol. 18, no. 5, 2018.

[83] Y. Yu, L. Ru, and K. Fang, "Bio-inspired mobility prediction clustering algorithm for ad hoc UAV networks." *Eng. Lett.*, vol. 24, no. 3, 2016.

[84] S. Mirjalili, "Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Comput. and Appl.*, vol. 27, no. 4, pp. 1053–1073, May 2016.

[85] A. Khan, F. Aftab, and Z. Zhang, "Bicsf: Bio-inspired clustering scheme for fanets," *IEEE Access*, vol. 7, pp. 31 446–31 456, 2019.

[86] H. Okcu and M. Soyturk, "Distributed clustering approach for UAV integrated wireless sensor networks," *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 15, pp. 106–120, 2014.

[87] M. Y. Arafat and S. Moh, "Routing protocols for unmanned aerial vehicle networks: A survey," *IEEE Access*, vol. 7, pp. 99 694–99 720, 2019.

[88] A. Jimenez-Pacheco, D. Bouhired, Y. Gasser, J. Zufferey, D. Floreano, and B. Rimoldi, "Implementation of a wireless mesh network of ultra light mavs with dynamic routing," in *Proc. IEEE GLOBECOM Wkshps*, Dec 2012, pp. 1591–1596.

[89] N. Farmani, L. Sun, and D. J. Pack, "A scalable multitarget tracking system for cooperative unmanned aerial vehicles," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 53, no. 4, pp. 1947–1961, Aug 2017.

[90] C. Zang and S. Zang, "Mobility prediction clustering algorithm for UAV networking," in *Proc.IEEE GLOBECOM Wkshps*, Dec 2011, pp. 1158–1161.

[91] M. Harounabadi, A. Puschmann, O. Artemenko, and A. Mitschele-Thiel, "TAG: Trajectory aware geographical routing in cognitive radio ad hoc networks with UAV nodes," in *Ad Hoc Networks*. Springer, 2015, pp. 111–122.

[92] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Unmanned aerial vehicle with underlaid device-to-device communications: Performance and tradeoffs," *IEEE Trans. Wireless Commun.*, vol. 15, no. 6, pp. 3949–3963, June 2016.

[93] C. A. Balanis, *Antenna theory: analysis and design*. John wiley & sons, 2016.

[94] A. Al-Hourani, S. Kandeepan, and A. Jamalipour, "Modeling air-to-ground path loss for low altitude platforms in urban environments," in *Proc. IEEE GLOBECOM*, Dec 2014, pp. 2898–2904.

[95] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage," *IEEE Commun. Lett*, vol. 20, no. 8, pp. 1647–1650, 2016.

[96] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," *IEEE Commun. Lett.*, vol. 3, no. 6, pp. 569–572, Dec 2014.

[97] E. Kalantari, H. Yanikomeroglu, and A. Yongacoglu, "On the number and 3d placement of drone base stations in wireless cellular networks," in *Proc. VTC-Fall*, Sep. 2016, pp. 1–6.

[98] A. French, M. Mozaffari, A. Eldosouky, and W. Saad, "Environment-aware deployment of wireless drones base stations with google earth simulator," in *Proc. IEEE PerCom Wkshps*, March 2019, pp. 868–873.

[99] R. El Mezouary, A. Choukri, A. Kobbane, and M. El Koutbi, "An energy-aware clustering approach based on the k-means method for wireless sensor networks," *Advances in Ubiquitous Networking*, pp. 325–337, 2016.

[100] M. Monwar, O. Semiari, and W. Saad, "Optimized path planning for inspection by unmanned aerial vehicles swarm with energy constraints," in *Proc. IEEE GLOBECOM*, 2018, pp. 1–6.

[101] Y. Ji, C. Dong, X. Zhu, and Q. Wu, "Fair-energy trajectory planning for multi-target positioning based on cooperative unmanned aerial vehicles," *IEEE Access*, vol. 8, pp. 9782–9795, 2020.

[102] O. Narmanlioglu and E. Zeydan, "Mobility-aware cell clustering mechanism for self-organizing networks," *IEEE Access*, vol. 6, pp. 65 405–65 417, 2018.

[103] G. V. Rossi, Z. Fan, W. H. Chin, and K. K. Leung, "Stable clustering for ad-hoc vehicle networking," in *Proc. IEEE WCNC*, March 2017, pp. 1–6.

[104] S. Bhandari, X. Wang, and R. Lee, "Mobility and location-aware stable clustering scheme for uav networks," *IEEE Access*, vol. 8, pp. 106 364–106 372, 2020.

[105] E. T. Michailidis, N. I. Miridakis, A. Michalas, E. Skondras, and D. J. Vergados, "Energy optimization in dual-ris uav-aided mec-enabled internet of vehicles," *Sensors*, vol. 21, no. 13, 2021.

[106] M. Mozaffari, W. Saad, M. Bennis, Y. Nam, and M. Debbah, "A tutorial on uavs for wireless networks: Applications, challenges, and open problems," *IEEE Commun. Surv. Tut.*, vol. 21, no. 3, pp. 2334–2360, thirdquarter 2019.

[107] J. Angjo, I. Shayea, M. Ergen, H. Mohamad, A. Alhammadi, and Y. I. Daradkeh, "Handover management of drones in future mobile networks: 6g technologies," *IEEE Access*, vol. 9, pp. 12 803–12 823, 2021.

[108] F. Zhou, R. Q. Hu, Z. Li, and Y. Wang, "Mobile edge computing in unmanned aerial vehicle networks," *IEEE Wirel. Commun.*, vol. 27, no. 1, pp. 140–146, 2020.

[109] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 2017.

[110] Y. Hui, Z. Su, and T. H. Luan, "Collaborative content delivery in software-defined heterogeneous vehicular networks," *IEEE/ACM Trans. Netw*, vol. 28, no. 2, pp. 575–587, 2020.

[111] H. Peng, Q. Ye, and X. Shen, "Spectrum management for multi-access edge computing in autonomous vehicular networks," *EEE Trans. Intell. Transp. Syst.*, vol. 21, no. 7, pp. 3001–3012, 2020.

[112] Y. Cai, L. Ran, J. Zhang, and H. Zhu, "Latency optimization for d2d-enabled parallel mobile edge computing in cellular networks," *EURASIP J Wirel Commun Netw*, vol. 2021, no. 1, pp. 1–23, 2021.

[113] A. Al-Shuwaili and O. Simeone, "Energy-efficient resource allocation for mobile edge computing-based augmented reality applications," *IEEE Commun. Lett*, vol. 6, no. 3, pp. 398–401, 2017.

[114] J. Xu, L. Chen, and S. Ren, "Online learning for offloading and autoscaling in energy harvesting mobile edge computing," *IEEE Trans. Cogn. Commun.*, vol. 3, no. 3, pp. 361–373, 2017.

[115] S. Bi and Y. J. Zhang, "Computation rate maximization for wireless powered mobile-edge computing with binary computation offloading," *IEEE Trans. Wirel. Commun.*, vol. 17, no. 6, pp. 4177–4190, 2018.

[116] W.-T. Li, M. Zhao, Y.-H. Wu, J.-J. Yu, L.-Y. Bao, H. Yang, and D. Liu, "Collaborative offloading for uav-enabled time-sensitive mec networks," *EURASIP J Wirel Commun Netw*, vol. 2021, no. 1, pp. 1–17, 2021.

[117] Z. Yu, Y. Gong, S. Gong, and Y. Guo, "Joint task offloading and resource allocation in uav-enabled mobile edge computing," *IEEE Internet of Things J.*, vol. 7, no. 4, pp. 3147–3159, 2020.

[118] S. Li, Q. Yu, M. A. Maddah-Ali, and A. S. Avestimehr, "A scalable framework for wireless distributed computing," *IEEE/ACM Trans. Netw*, vol. 25, no. 5, pp. 2643–2654, 2017.

[119] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation rate maximization in uav-enabled wireless-powered mobile-edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1927–1941, 2018.

[120] N. H. Motlagh, M. Bagaa, and T. Taleb, "Uav-based iot platform: A crowd surveillance use case," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 128–134, 2017.

[121] S. Jeong, O. Simeone, and J. Kang, “Mobile edge computing via a uav-mounted cloudlet: Optimization of bit allocation and path planning,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 3, pp. 2049–2063, 2018.

[122] H. Peng, Q. Ye, and X. Shen, “Spectrum management for multi-access edge computing in autonomous vehicular networks,” *EEE Trans. Intell. Transp. Syst.*, vol. 21, no. 7, pp. 3001–3012, 2020.

[123] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, “Computation rate maximization in uav-enabled wireless-powered mobile-edge computing systems,” *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1927–1941, 2018.

[124] C. Zhan and Y. Zeng, “Completion time minimization for multi-uav-enabled data collection,” *IEEE Trans. Wirel. Commun.*, vol. 18, no. 10, pp. 4859–4872, 2019.

[125] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, “Mobile unmanned aerial vehicles (uavs) for energy-efficient internet of things communications,” *IEEE Trans. Wirel. Commun.*, vol. 16, no. 11, pp. 7574–7589, 2017.

[126] W. Ding, Z. Yang, M. Chen, J. Hou, and M. Shikh-Bahaei, “Resource allocation for uav assisted wireless networks with qos constraints,” in *Proc. IEEE WCNC*, 2020, pp. 1–7.

[127] P. V. Klaine, J. P. Nadas, R. D. Souza, and M. A. Imran, “Distributed drone base station positioning for emergency cellular networks using reinforcement learning,” *Cogn Comput*, vol. 10, no. 5, pp. 790–804, 2018.

[128] M. Lee, G. Yu, and G. Y. Li, “Graph embedding based wireless link scheduling with few training samples,” *IEEE Trans. Wirel. Commun.*, pp. 1–1, 2020.

[129] G. Li, Q. Lin, J. Wu, Y. Zhang, and J. Yan, “Dynamic computation offloading based on graph partitioning in mobile edge computing,” *IEEE Access*, vol. 7, pp. 185 131–185 139, 2019.

[130] C. You, K. Huang, H. Chae, and B. Kim, “Energy-efficient resource allocation for mobile-edge computation offloading,” *IEEE Trans. Wirel. Commun.*, vol. 16, no. 3, pp. 1397–1411, 2017.

[131] T. X. Tran and D. Pompili, “Joint task offloading and resource allocation for multi-server mobile-edge computing networks,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 856–868, 2019.

[132] A. Paris, H. Mirghasemi, I. Stupia, and L. Vandendorpe, “Leveraging user-diversity in energy-efficient edge-facilitated collaborative fog computing,” *IEEE Access*, vol. 9, pp. 95 636–95 650, 2021.

[133] ——, “Energy-efficient edge-facilitated wireless collaborative computing using map-reduce,” in *Proc. IEEE SPAWC Wkshps*, 2019, pp. 1–5.

Curriculum Vitae

Name: Sabin Bhandari

Post-Secondary Education and Degrees:

2016 - Present, Ph.D.
Electrical and Computer Engineering
The University of Western Ontario
London, Ontario, Canada

2014 - 2016, M.Eng
Computer Engineering
Chosun University
Gwangju, South Korea

2006 - 2010, B.Eng
Electronics and Communication Engineering
Kantipur Engineering College, Tribhuvan University
Kathmandu, Nepal

Honours and Awards: 2020, Outstanding Young Professional Award, IEEE London Section

Related Work Experience:

Research Assistant
The University of Western Ontario, London, Ontario, Canada
2016 - Present

Teaching Assistant
The University of Western Ontario, London, Ontario, Canada
2016 - 2021

Research Assistant
Chosun University, Gwangju, South Korea
2014-2016

Lecturer
Kantipur Engineering College, Kathmandu, Nepal
2011-2014

Publications:

1. **S. Bhandari**, S. K. Sharma, X. Wang,“Cloud-assisted device clustering for lifetime prolongation in wireless IoT networks,” *CCECE 2017 -2017 IEEE Canadian Conference on Electrical and Computer Engineering* , pp. 1-4, April 2017.

2. **S. Bhandari**, S. K. Sharma, X. Wang, “Latency minimization in wireless IoT using prioritized channel access and data aggregation,” *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, pp. 1-6, December 2017.

3. **S. Bhandari**, S. K. Sharma, X. Wang, “Device Grouping for Fast and Efficient Channel Access in IEEE 802.11ah based IoT Networks,” *ICC Workshops 2018 - 2018 IEEE International Conference on Communications Workshops*, pp. 1-6, May 2018.

4. **S. Bhandari**, X. Wang, “Prioritized Clock Synchronization for Event Critical Applications in Wireless IoT Networks,” *IEEE Sensors Journal*, vol. 19, no. 16, pp. 7120-7128, August 2019.

5. **S. Bhandari**, X. Wang, Richard Lee “Mobility and Location-Aware Stable Clustering Scheme for UAV Networks,” *IEEE Access*, vol. 8, no. 16, pp. 106364-10637, 2020.

6. **S. Bhandari**, X. Wang “Intelligent Resource Allocation Scheme for UAV Swarm Network,” *ICC 2022 - 2022 IEEE International Conference on Communications* (Submitted).

7. **S. Bhandari**, X. Wang “QoS aware Edge-Facilitated Wireless Collaborative Computing in UAV Networks,” *IEEE Transactions on Industrial Informatics* (To be submitted).