Prioritised Random Access Channel Protocols for Delay Critical M2M Communication over Cellular Networks

Mahmudul Haque Kafi

Doctor of Philosophy

University of York
Electronic Engineering

May 2021

Abstract

With the ever-increasing technological evolution, the current and future generation communication systems are geared towards accommodating Machine to Machine (M2M) communication as a necessary prerequisite for Internet of Things (IoT). Machine Type Communication (MTC) can sustain many promising applications through connecting a huge number of devices into one network. As current studies indicate, the number of devices is escalating at a high rate. Consequently, the network becomes congested because of its lower capacity, when the massive number of devices attempts simultaneous connection through the Random Access Channel (RACH). This results in RACH resource shortage, which can lead to high collision probability and massive access delay. Hence, it is critical to upgrade conventional Random Access (RA) techniques to support a massive number of Machine Type Communication (MTC) devices including Delay-Critical (DC) MTC. This thesis, approaches to tackle this problem by modeling and optimising the access throughput and access delay performance of massive random access of M2M communications in Long-Term Evolution (LTE) networks. This thesis investigates the performance of different random access schemes in different scenarios. The study begins with the design and inspection of a group based 2-step Slotted-Aloha RACH (SA-RACH) scheme considering the coexistence of Human-to-Human (H2H) and M2M communication, the latter of which is categorised as: Delay-Critical user equipments (DC-UEs) and Non-Delay-Critical user equipments (NDC-UEs). Next, a novel RACH scheme termed the Priority-based Dynamic RACH (PD-RACH) model is proposed which utilises a coded preamble based collision probability model. Finally, being a key enabler of IoT, Machine Learning, i.e. a Q-learning based approach has been adopted, and a learning assisted Prioritised RACH scheme has been developed and investigated to prioritise a specific user group. In this work, the performance analysis of these novel RACH schemes show promising results compared to that of conventional RACH.

Table of contents

List of figures

List of tables

Glossary

AI	Artificial Intelligence
AC	baring access class
ACB	Access Class Baring
BI	back-off indicator
BS	Base Station
CCCH	Common Control Channel
CDF	cumulative density function
CP	Cyclic Prefix
CRA	Coded Random Access
DC-UEs	Delay-critical User-Equipment
DCCH	Dedicated Control Channel
DP	Dynamic Programming
DTCH	Dedicated Traffic Channel
EAB	Extended access barring
eMBB	enhanced Mobile Broadband
eNB	Evolved Node B
eNodeB	Evolved Node B
ETSI	European Telecommunications Standard Institute
GOS	Grade of Service
GSM	Global System for Mobile Communications
HSPA	High Speed Packet Access
HTC	Human Type Communication
H2H	Human-to-Human
IoT	Internet of Things
ITU-R	International Telecommunication Union- Radio communication Sector
LTE	Long Term Evolution
LTE-A	Long Term Evolution Advanced
MAC	Medium Access Control
MDP	Markov Decision Process
MIMO	Multiple Input Multiple Output
ML	Machine Learning
MTC	Machine Type Communication
mMTC	massive Machine Type Communications
M2M	Machine-to-Machine
NDC-UEs	Non-Delay-critical User-Equipment
NOMA	Non-Orthogonal Multiple Access

OFDM	orthogonal frequency-division multiplexing
OPNET	Optimized Network Engineering Tools
PD-RACH	Priority-based Dynamic RACH
PDSCH	Physical Downlink Shared Channel
PrachConfigIndex	PRACH Configuration Index
PRACH	Physical Random Access CHannel
PUCCH	Physical Uplink Control Channel
PUSCH	Physical Uplink Shared CHannel
QL-RACH	Q-learning assisted Random Access Channel
QoS	Quality of Service
QPD-RACH	Q-learning assisted Prioritised Dynamic RACH
RA	Random Access
RACH	Random Access Channel
RAN	Radio Access Network
RAO	Random Access Opportunities
RAR	Random Access Response
RB	Resource Blocks
RL	Reinforcement Learning
RRC	Radio Resource Control
RRC-CONNECTED	Radio Resource Control Connected
RRC-IDLE	Radio Resource Control Idle
SA	slotted-ALOHA
s-ALOHA	slotted-ALOHA
SA-RACH	slotted-ALOHA Random Access Channel
SCMA	Sparse Code Multiple Access
SF	System Frame
SFN	System Frame Number
SIB2	System Information Block2
SIM	Subscriber Identity Module
Tb	baring time duration
TD	Temporal Difference
UCI	Uplink Control Information
UE	User-Equipment
UL	Uplink
UL-SCH	Uplink Shared Channel
URLLC	Ultra-Reliable and Low Latency Communications
WCDMA	Wideband Code Division Multiple Access
WiMAX	Worldwide Interoperability for Microwave Access
1G	First Generation
2G	Second Generation
3G	Third Generation
3GPP	3rd Generation Partnership Project
4G	Fourth Generation
5G	Fifth Generation
6G	Sixth Generation

Symbols

C	data rate
D	average delay
$F(t)$	cumulative density function
G	average call/packet arrivals in a given time using predicted traffic distribution
G_u	offered traffic level of an individual user
L	number of successful transmissions
m	number of servers
PB	blocking probability
P_b	predicted allowed blocking probability
RB	total number of blocked RACH requests
R_s	number of successful RACH requests
$S(E)$	throughput
t_g	time of RACH request generation
t	inter inter-arrival time between the successful calls or RACH requests
T_l	generated traffic load
t_s	time of successful RACH transmission
t_total	the period over which the throughput is calculated
T_tx	transmission time of each packet
λ	average number of service requests per unit time
τ	average service duration
l	Packet length

In the name of god, the most merciful and benevolent. All glory to God for his graces for giving me the strength and perseverance to carry on with my work even at the most difficult times. I would like to dedicate this thesis to my loving parents and wife. . . .

Acknowledgements

First and foremost, thank be to God for blessing me for this outstanding opportunity and guiding me throughout the whole journey.

I would like to express my sincere and deepest gratitude to both my supervisors, Prof. Paul D. Mitchell and Prof. David Grace for their persistent and invaluable guidance, exceptional encouragement, motivation, and constant support. Their numerous suggestions on different research directions, boundless freedom to explore them, and constructive feedback helped me pass through all the difficulties during my PhD study.

Also, I would like to thank my thesis advisor, Prof. Alister Burr for his invaluable suggestions and comments during the periodic research assessments that helped in shaping the research work.

I must also convey my heartfelt thanks to my undergraduate teacher and mentor Prof. Shaikh Enayet Ullah, who introduced me to the field of research and assisted me in laying the foundation for my research career.

Thanks to all my previous and current colleagues at the Department of Electronics for fostering a supportive research environment, in particular Wael, Tariq, Ibrahim, Shammi, Mahama, and Shipra. Many thanks to all my friends in York especially Vishwadeep, Omkar, Rabbani, and Arnold for making my life in York more enjoyable.

Finally, I am grateful to my entire family especially my parents for their endless love and prayers, my sisters for their understanding, support, and advise in every step of my journey. I am truly thankful to my wife for her love, friendship, and care, which has made my daily life more pleasant and the PhD path less difficult. In the end, I would like to mention my friends in Bangladesh in particular Pias, Sourov, Bishwajit, Rini for their belief, support and love for me even when I was so far away.

Declaration

To the best of my knowledge and belief, I declare that this thesis is original and the work described in it has been composed solely by me. The appropriate references and acknowledgements to other researchers has been included in this work. I certify that this work does not contain any material that has been previously approved or submitted for the award of any other degree, in whole or in part. Part of this work have resulted in the publications and awards listed below:

- (Conference) M. H. Kafi, P. Mitchell and D. Grace, "Prioritised Dynamic RACH (PD-RACH) Scheme for Delay-Critical MTC Communication," 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, pp. 1-6, (2020).

- (Conference) M. H. Kafi, P. Mitchell and D. Grace, "Q-learning assisted Coded Preamble based Prioritized Dynamic RACH for Delay-Critical MTC," to be submitted to IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (2021).

- (Article) M. H. Kafi, P. Mitchell and D. Grace, "Learning-assisted Dynamic Prioritised Random Access Scheme for mMTC in Cellular IoT network," to be submitted to IEEE Access, (2022).

- (Poster prize) 3rd Prize Poster Competition. Second Year PhD Poster Competition, Electronic Engineering Department, University of York.

Mahmudul Haque Kafi
May 2021

Chapter 1

Introduction

1.1 Overview

In recent years, the Internet of Things (IoT) has experienced a radical shift with the ever-increasing number of devices switching from wired to wireless networks as well as with the considerable increase in data rates [1]. Due to this paradigm shift towards IoT, the number of Machine-Type Communications (MTC) is increasing rapidly and thus MTC is a crucial target scenario in cellular networks such as 4G, 5G and beyond [2]. As of today, many countries have implemented the fifth-generation (5G) mobile communication system designed to address more complex Quality of Service (QoS) requirements than its predecessors. These requirements emerged from a wide range of applications and services that were previously non-existent. The ITU-R has recommended three usage scenarios for 5G and beyond cellular networks [3]:

- enhanced Mobile Broadband (eMBB),
- massive Machine Type Communications (mMTC), and
- Ultra-Reliable and Low Latency Communications (URLLC).

The eMBB prioritises wide-area coverage to deliver smooth access and high capacity in hotspots and targets human-centric applications requiring high-data-rate accessibility to mobile services, multi-media content, and data. URLLC enables reliable, fast and ubiquitous connectivity for emerging mission-critical applications (e.g. self-driving cars, smart grids, and industry 4.0) which have strict requirements for their seamless operation. On the contrary, mMTC intends to connect tens of billions of network-enabled devices wirelessly with little to no human intervention [4]. Additionally, mMTC facilitates a wide variety of new services

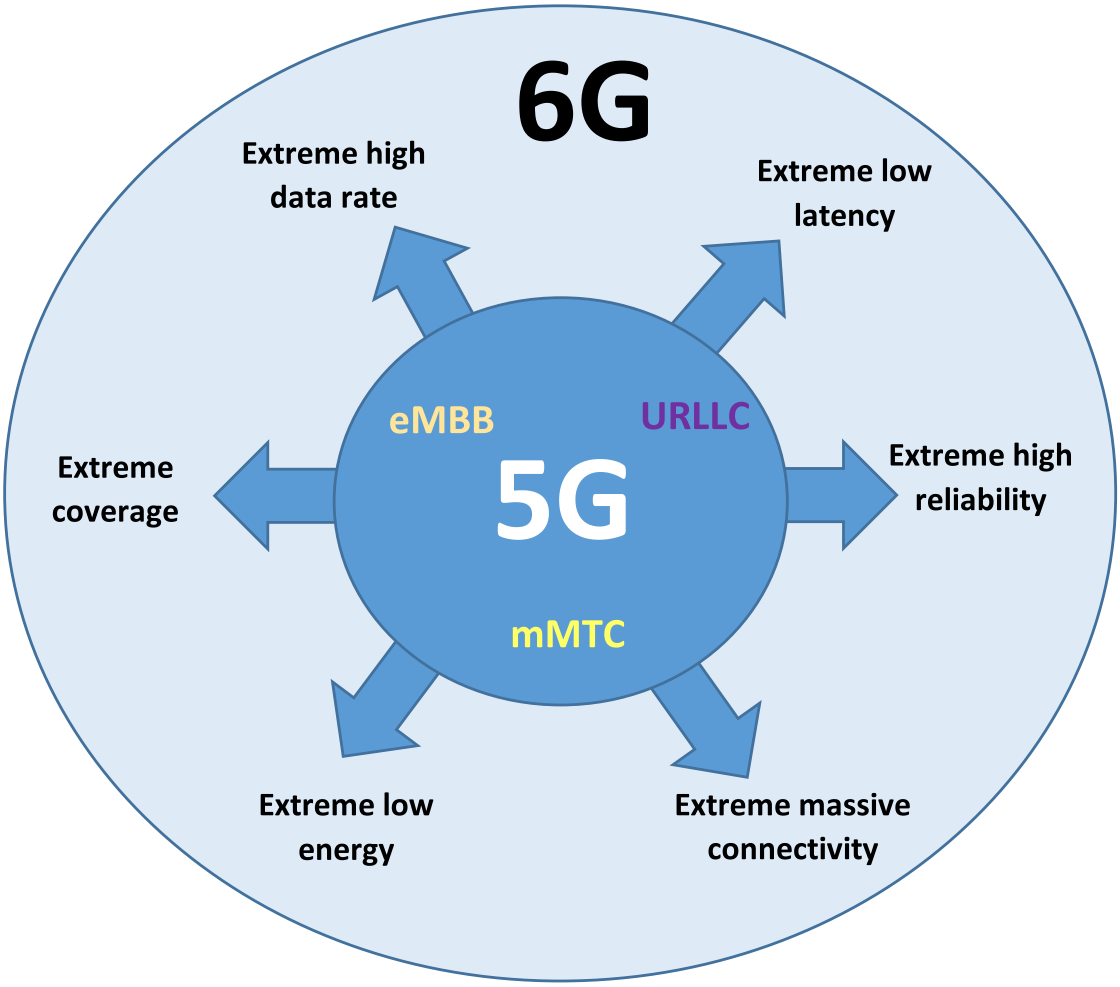

Figure 1.1: Requirements for 5G and 6G wireless technology

and applications such as industrial automation, intelligent transportation, e-health, smart grid etc. [5].

Some key traffic characteristics of mMTC include very small transmit-data size per device, exceptionally high-energy efficiency requirement, majorly uplink (UL), partially/fully autonomous communication, and most importantly sporadic transmission [6].

Fig. 1.1 shows the requirements of the ongoing 5G as well as next-generation 6G technology [7], where, in addition to the higher 5G criteria, new standards that were not addressed in 5G have been recommended. 6G also opens the door to plenty of new application possibilities such as (i) human-centric services, (ii) long-distance and high-mobility communications, (iii) extremely low-power communications, (iv) convergence of communications, computing, control, localization, and sensing, (v) space-air-ground sea integrated networks, (vi) distributed AI (e.g., federated learning) applications, (vii) remote holographic unmanned systems to name a few [7] and [8].

Although conventional cellular systems (5G, LTE/LTE-A) have many advantages in terms of large coverage, time synchronisation, and handover, when it comes to supporting a massive number of devices, mMTC poses some additional challenges like QoS provisioning, handling highly dynamic and sporadic MTC traffic, massive signalling overhead, congestion and network overload in the physical random access channel (PRACH) [9]. Using PRACH, mMTC and H2H devices perform an initial access request procedure called Random Access (RA) process. In the current RA procedure in the LTE/LTE-A system, device sends a uniformly selected preamble on the RACH to establish a link to Evolved Node B (eNB). However, due to the limited number of available preambles, when a large number of devices are attempting RA within a short interval of time, it causes an access problem referred to as the "massive access problem" as mentioned by the 3rd Generation Partnership Project (3GPP). As a result, while using conventional RA procedure in a massive access scenario, there will be RACH resource shortage, which can result in high collision probability and massive access delay [10]. Therefore, to support a large number of devices as well as URLLC based MTC devices mentioned above, it is crucial to improve conventional RA methods.

This thesis, therefore, considers and investigates the coexistence of H2H and MTC devices, where MTC devices are categorized into two groups: delay-critical user equipments and non-delay-critical user equipments. The main motivation is on how to more effectively handle machine type traffic over the RA channel, including how to effectively prioritise delay-critical traffic and provide the required quality of service without degrading the performance of existing H2H traffic.

1.2 Hypothesis

The following hypothesis guides the research work provided in this thesis:

"A Dynamic Prioritised Random Access strategy can provide priority access to a specified user group termed Delay-critical UEs, by minimising access delay and providing required Quality of Service (QoS) in a congested network scenario caused by a massive number of machine-type devices."

Future generation mobile communication system standards (5G and beyond) will employ a range of techniques geared towards supporting machine to machine communication, as required by the IoT. Due to this shift towards IoT, the number of machine-type communications (MTC) is increasing rapidly. When a large number of MTC devices send an uplink signalling message through the RACH in a short interval of time, network congestion occurs. As a consequence, there will be a RACH resource shortage, which can result in high collision probability and massive access delay. Therefore, to support a large number of MTC devices including delay-critical MTC, it is crucial to improve conventional RA methods.

This thesis, therefore, considers mMTC scenario and investigates the coexistence of H2H and M2M devices, where M2M devices are categorised into two groups: Delay-critical user equipment (DC-UEs) and non-Delay-critical user equipment (NDC-UEs). To mitigate these issues, a dynamic RA scheme with prioritisation will be needed, where eNB can handle massive access requests by dynamically controlling the devices, as well as providing required priority access to the specific user group to provide the required quality of service. In this work, a two-step RA scheme is considered using a coded preamble technique in order to prioritise RA schemes dynamically. Additionally, we have considered Q-learning techniques which is a basic model of Reinforcement Learning (RL). Here, the idea is to avoid collision by learning in a contention-based RA process which will provide priority to a specific user group.

1.3 Thesis Structure

The remainder of the thesis is organised into seven chapters for the major body. The topics of each chapter are briefly summarised in this section. Fig. 1.2 visually depicts the relationship between them and the relevant completed/in-progress publications. The arrows represent the flow of information between complementary chapters, while the boxes highlight crossed chapters where the information from one chapter is needed to properly comprehend the material in the others.

Chapter 2 provides an overview of the concept of M2M communication including its architecture, enabling technologies, applications and requirements. As a part of the literature review, research on supporting M2M on cellular networks are discussed. Finally, different machine learning techniques and their benefits for M2M communication are presented.

Chapter 3 provides background information on LTE cellular network including its network structure, frame structure, and channel configuration. The chapter also reviews Random Access Channel, providing the RA procedures, the impact of M2M communication on RACH, limitation on RACH, and RACH overload control mechanism due to its relevance in this work.

Chapter 4 begins by introducing the methods used to develop the simulation models for an uplink cellular system. First, an overview of the simulation software tools used in this thesis work is discussed. Next, the performance metrics that are used to evaluate network performance are defined. Finally, the traffic model and validation techniques are presented.

Chapter 5 introduces a two-step RACH scheme using slotted-ALOHA (SA), where a combination of H2H and M2M traffic is considered. First, an analytical model to develop RACH-throughput performance is presented. Next, the cause of RACH instability is illustrated by analysing the performance of two steps RACH-throughput, average end-to-end delay and blocking probability using retransmission limits and back-off interval windows.

Chapter 6 presents a scenario, where a shared RACH Channel with the coexistence of H2H and M2M UEs is considered and M2M users are categorised into two groups: delay-critical (DC-UEs) and non-delay-critical (NDC-UEs). It further describes the implementation of a novel RACH scheme called the Priority-based Dynamic RACH (PD-RACH) using a two-step SA-RACH approach to control the interaction and the delay for the specific delay-critical user group. A coded preamble based collision probability model, which provides a two-step prioritisation scheme is also discussed.

Chapter 7 introduces a novel Q-learning assisted Prioritised Dynamic RACH (QPD-RACH) scheme for delay critical user equipments; an extension to the PD-RACH approach that was proposed in chapter 6. This approach introduces Q-learning in order to avoid collision among delay critical UEs to minimise delay and provide priority access to delay-

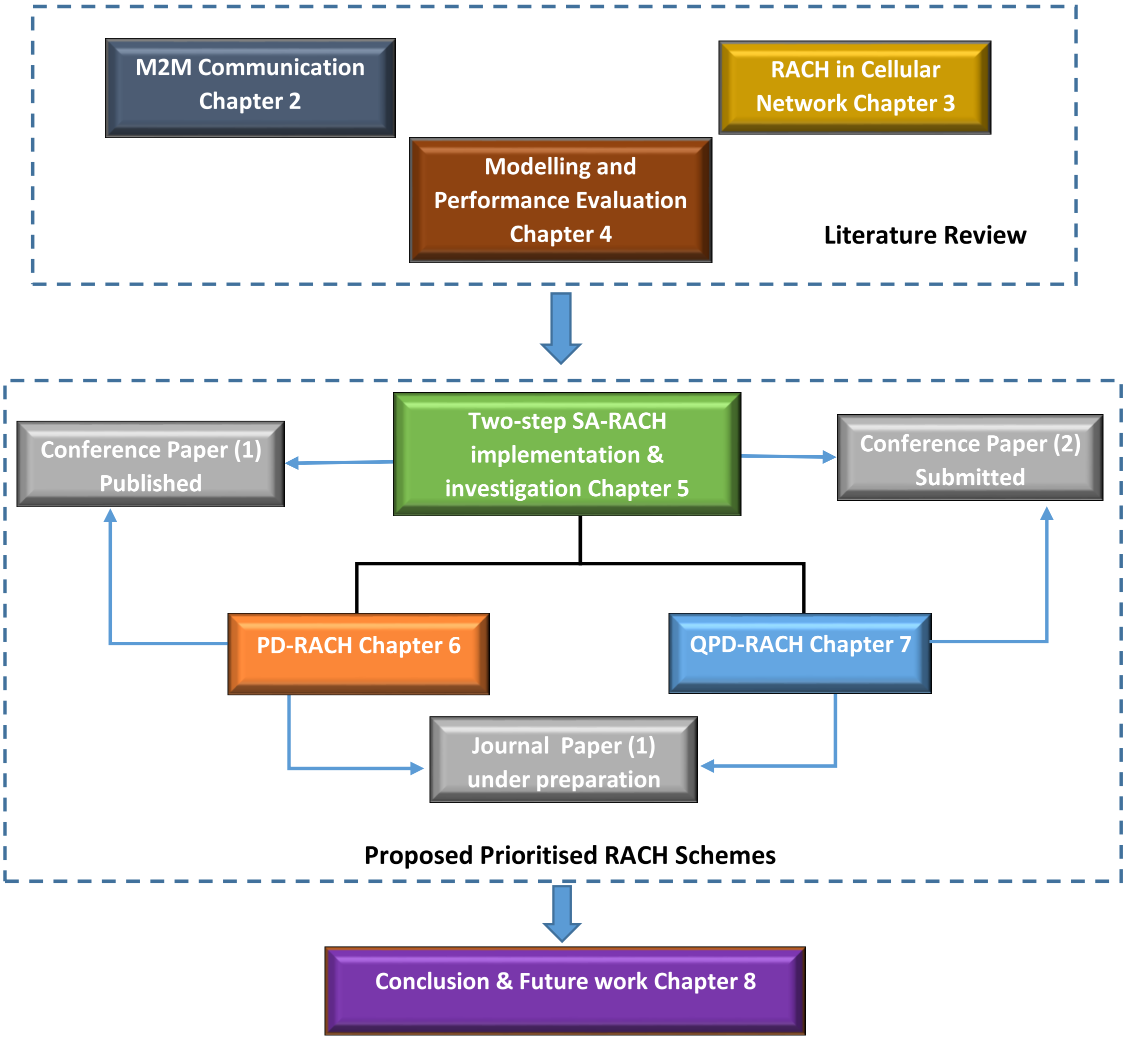

Figure 1.2: Block diagram presentation of thesis structure with proposed RACH Schemes

critical UEs over non-delay-critical UEs in an emergency scenario. A scenario of a combined RACH access scheme is presented, where H2H traffic uses slotted-ALOHA RACH access (SA-RACH) and M2M traffic controlled by the Q-Learning assisted RACH (QL-RACH). Finally, simulations results of the proposed scheme are presented.

Chapter 8 summarises the work presented in this thesis, concludes its original contributions, and suggests a number of ideas for further work.

Chapter 2

Machine-to-Machine Communication

2.1 Introduction

The objective of this chapter is to provide background information about key elements related to this thesis including a brief description of related topics in the literature. In section 2.2, we introduce concepts of Machine-to-Machine (M2M) communication including its architecture, enabling technologies, applications and requirements. Section 2.3 presents M2M in cellular networks, describing advantages and challenges. In section 2.4, we briefly discuss machine learning techniques and its advantages for M2M communication. Finally, we summarise the chapter in section 2.5.

2.2 Machine-to-Machine Communications

In modern wireless communication, Machine-to-Machine (M2M) communication is a rapidly evolving field. In this form of communication, one or more entities (e.g. "intelligent devices") can communicate autonomously with little or no human intervention. Such autonomous communication includes a diverse set of actions including sensing, actuation, monitoring, and processing etc. These smart devices are on the rise and are predicted to reach around 125 billion in number by 2030 [1]. A number of factors contribute to this current escalation in device numbers, including inexpensive electronic gadgets, sensors, actuators, and embedded devices, abundance of high-speed internet access, and the use of automated devices in remote spots. 3GPP refers to M2M communication as machine-type communication (MTC) and started standardisation activities on MTC in September 2008 as part of 3GPP Rel-10 specifications [19]. Whereas M2M/MTC needs an infrastructure to operate, device to device communication (D2D), on the other hand, allows UEs to communicate with each other

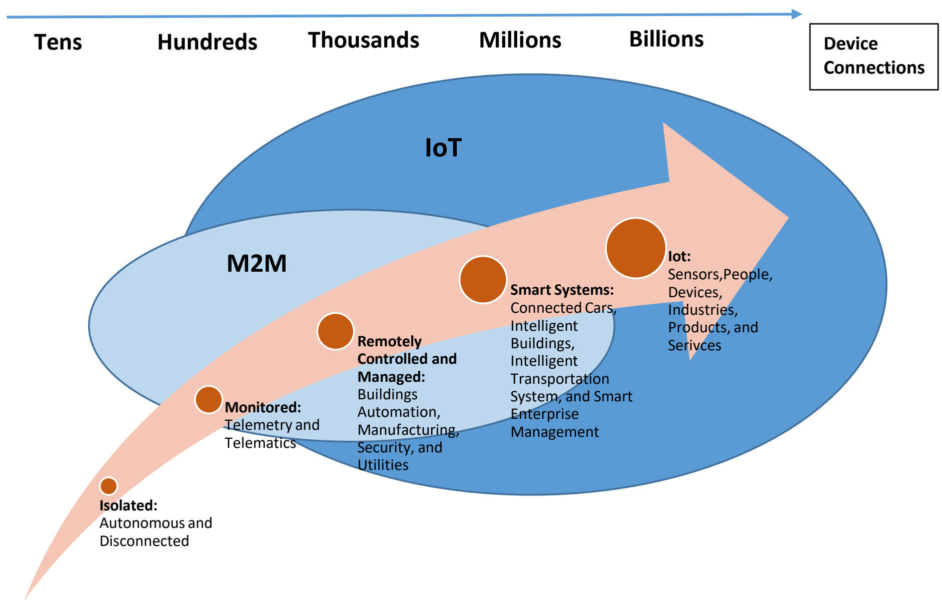

Figure 2.1: Evolution of IoT (reproduced from [21])

without the involvement of network infrastructure such as access points or BS. In our work, we focused on M2M/MTC communication and we provide a more detailed overview of M2M communication in later sections.

2.2.1 M2M evolution towards IoT

Next-generation wireless communication systems are rapidly evolving and one of the key enablers of these systems is the Internet of Things (IoT) [12]. In an IoT ecosystem, "objects/things" are integrated with machines, sensors, software, and network connectivity in a way that they can collect and exchange data autonomously. This innovative technology allows more direct integration between the physical world and computer-based systems, resulting in enhanced efficiency, accuracy and economic advantage [13]. In a nutshell, IoT is facilitated by integrating a massive number of M2M devices using the internet to process all the data. That is why M2M communication is considered as an integral part of IoT. Fig.2.1 depicts the evolution of IoT from M2M communication and shows how M2M plays a vital role in IoT evolution as a fundamental building block.

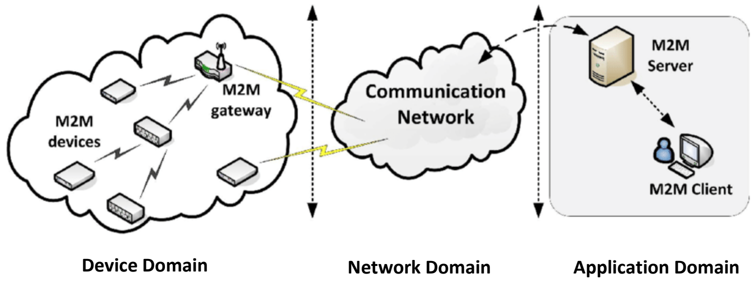

Figure 2.2: M2M system model (reproduced from [22])

2.2.2 M2M architecture and Enabling Technologies

M2M Architecture: According to the European Telecommunications Standard Institute (ETSI) architecture [14], the M2M communication system model comprises of three interlinked domains namely the M2M device domain, the network domain, and the application domain. Fig.2.2 shows the M2M system model which is explained below.

- M2M device domain: In this domain, a large number of devices (e.g. sensors, actuators, and smart meters) and gateways (e.g. data aggregation points) collaborate with each other, collect and exchange their information by creating an M2M area network. The types and numbers of devices in this domain vary depending on their individual application scenarios.

- Network domain: This domain creates links between the M2M device domain and the application domain using long-range wired/wireless network protocols such as WiMAX, 3G, 4G/LTE, 5G and beyond cellular networks. These networks provide cost-efficient and reliable channels with wide coverage in order to transfer sensory information from the device domain to the application domain.

- Application domain: This domain has two core features, the first is called the M2M server which processes the collected data and interacts with the M2M devices, and the second one is the M2M client which is used to serve users.

Enabling technologies: In order to establish a connection between domains, M2M adopts various technologies that can be achieved through both wired and wireless technologies. Although, wired M2M communication is able to provide high data rates and be extremely reliable, it will be very expensive to roll out for huge number of devices, causing difficulty of implementation. Therefore, wired communication is not further considered in our work. On the contrary, implementation of M2M communication is possible using

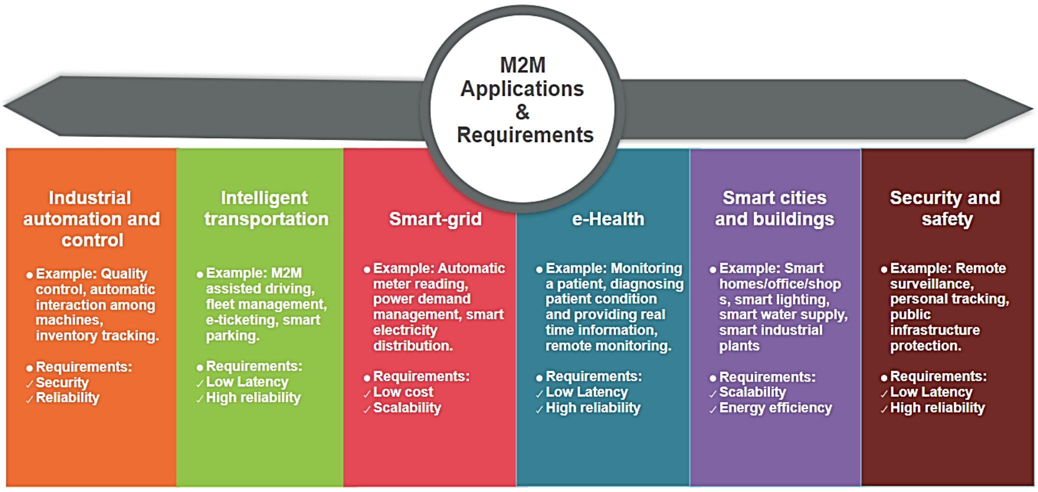

Figure 2.3: M2M Applications and Requirements

both cellular and non-cellular technologies. Cellular technologies such as GSM, WCDMA, LTE/LTE-A, 5G, and beyond can support M2M communication [15]. In this research, we mainly focus on the M2M access network which is directly related to the M2M device domain and network domain using cellular technologies.

2.2.3 Applications, Requirements and Features of M2M

With mMTC, a diverse range of new services and applications can be offered which include Industrial Automation and control, Intelligent Transportation, e-Health, Smart Grid, Environmental monitoring, Security and public safety and many more [2], [15], and [16]. Recent times have shown growth in the number of application cases with very different features and requirements that add constraints on the network technology as well as on MTC devices. In the following section, we discuss some of the key features and service requirements of some M2M applications [16], [19] shown in Fig.2.3.

Simultaneous massive transmission: Accommodating a massive number of devices concurrently on the network may be necessary for some applications, which may be achieved by enhancements in channel request, allocation protocols, as well as cooperative communication. In cooperative communication, nodes or terminals in a communication network collaborate with each other in information transmission to enable efficient utilization of communication resources [87].

Priority scheduling and access: These are essential to provide appropriate priority access through improved channel request and allocation protocols for certain types of devices, in a wide range of applications such as e-health, security surveillance etc.

Low power consumption: As most MTC devices are battery powered with power constraints, to reduce power consumption, signaling updates and an efficient sleep and wake up mechanism for the MTC devices may be required.

Bursty Traffic: M2M devices generate bursty natured traffic in most cases. To support this characteristic, changes may be required including new allocation protocols, channel coding and frame structure.

Monitoring and security: It is essential for M2M devices to be able to sense unusual activities, and enable proper authentication to prevent incidences such as hacking, that may compromise credentials and configurations. To improve monitoring and security, efficient mobility management mechanisms as well as interference mitigation mechanisms may be required.

Reliability: Irrespective of the working environment (e.g. mobility, interference, and channel quality) connection for M2M devices must be ensured, which is expressed as the reliability of the network. Reliability is strictly required for applications such as healthcare, remote payment etc.

Low Latency: Many M2M applications (e.g. healthcare, automated vehicles) demand low latency communication for which the network latency and data transmission latency both need minimising. This may necessitate changes to channel request and allocation protocols as well as changes in frame structure and control signaling.

Low or no mobility: For application cases where the M2M devices remain nearly stationary, changes to handover signaling and execution may be required for optimisation of mobility management to minimize power consumption as well as signaling overhead.

Based on delay requirements discussed above, M2M communication applications can be divided into two categories: (i) Delay-critical, applications needing low latency and (ii) Delay tolerant applications which are less sensitive to delays. Furthermore, different applications can have different priority based on their nature, and in our work, we are mainly focusing on delay-critical priority-based M2M communication.

2.3 M2M in Cellular Networks

Cellular networks (such as GSM, GPRS, LTE/LTE-Advanced and 5G) are the most widely known and used wireless network technology for provision of numerous services and ubiquity. Cellular networks can support M2M communication over a long-range network. The advantages as well as challenges of using a cellular network to support M2M communication are presented below.

2.3.1 Advantages of cellular network in M2M

Cellular networks are convenient for their high data rates, availability and ubiquity over the installation of a new private radio network [15]. This is an advantage for various M2M applications over other technologies, which are currently capable of supporting M2M communication. Moreover, lower cost for deployment of M2M can be achieved with cellular M2M communication. With cellular technology, an M2M device is able to establish direct cellular communication using its unique Subscriber Identity Module (SIM) card [17]. Additionally, cellular networks support roaming and mobility. One of the key advantage of cellular M2M lies in its ability to provide Quality of Service (QoS) guarantees. Considering all the advantages of cellular M2M communication over non-cellular options, such as Wi-Fi, Ethernet, Bluetooth etc., we opted for cellular M2M in our work.

2.3.2 Challenges in cellular network for M2M

Although cellular networks provide a wide variety of advantages, using a legacy cellular network is inefficient when the number of devices is enormous as it is mainly designed for human-centric communications termed Human-to-Human (H2H) communication [18]. High data rates, mobility, decent QoS for human satisfaction are some of the basic attributes of these cellular networks. On the contrary, M2M communication initiates and requires a very different set of characteristics and requirements (discussed in section 2.2.4) from that of regular H2H communication [19]. Table 2.1 summarises characteristic differences between H2H and M2M communications [9] and [20]. Some of the challenges faced in cellular network for M2M communications include: supporting the deployment of a massive number of MTC devices, accommodating small data bursts, ensuring a high level of security, providing ultra-reliable communications with low-latency, achieving low power consumption, supporting low-cost devices, enhancing coverage [5] etc.

In order to mitigate the challenges, substantial research and development effort has been undertaken by academia and industry.

Table 2.1: Characteristics differences between M2M and H2H

Characteristics	H2H	M2M
Density	Maybe not that much compared to M2M potential	M2M outnumbers human end users by an order of magnitude.
Downlink and Uplink	Most traffic is downlink and traditionally less traffic in Uplink	Uplink dominant traffic of small size, except for case of video surveillance.
Delay	250 ms (voice) to few second	10 ms to several minutes [4]
Mobility	Fixed to Low	Fixed to High
Traffic Transmission Frequency	Infrequent to Frequent	Infrequent
Priority Services	Yes	Yes
Power Consumption	High	Low
QoS Requirements	H2H users have similar service requirements	Different M2M applications may have diverse quality of service requirements.

2.3.3 Potential Enablers for designing cellular MTC

In addition to the existing solutions, some key enabling strategies are being studied for integrating mMTC into the cellular IoT network, some of which are briefly outlined below [2].

Dynamic resource allocation techniques: 3GPP has considered a possible solution to the RAN congestion problem through dynamic RACH resource allocation [35] with an aim to maximise its throughput, which can be done by selecting the number of preambles adaptively without knowing the number of devices or access probability [36]. One advantage of this method is reduced data collection time from resource-constrained MTC devices in delay-sensitive situations.

Clustering and data aggregation schemes: The RAN congestion may be greatly reduced by clustering MTC devices based on some relevant criteria (e.g. geographical locations or QoS requirements), and then aggregating the individual device data at the MTC gateway/aggregator [37]. Furthermore, investigating energy-efficient clustering techniques makes it easier to install low-power MTC devices.

Advanced RA schemes: Several developing RA schemes e.g. Non-Orthogonal Multiple Access (NOMA), Sparse Code Multiple Access (SCMA), Coded Random Access (CRA) [2], and distributed queuing based access protocol [38], are promising enablers for mMTC in cellular networks.

Low signalling overhead MAC protocols: Low signaling overhead MAC protocols target to reduce one of the primary technological problems created by MTC devices which is signaling overhead, and assist the deployment of cellular MTC [9].

Advanced transmission-scheduling techniques: In order to accommodate MTC devices with diverse QoS requirements, advanced transmission-scheduling technique is required. In this context, advance transmission-scheduling techniques such as delay-aware scheduling [39], fast uplink grant [40], and learning-assisted scheduling have the potential to handle and schedule a massive number of MTC over limited RACH resources.

In addition to the promising enablers discussed in the above section, it is worthwhile to mention machine learning (ML) techniques as a promising tool to aid mMTC in cellular IoT. In the following section, the advantages and types of ML techniques are discussed briefly.

2.4 Machine Learning for MTC in Cellular IoT

As recent times have experienced advancement in computing power as well as the ability to collect and store huge amounts of data, Machine Learning (ML) has found its way into many different scientific domains. ML is a powerful set of mathematical and computational tools that aim to acquire the system variations/parameters uncertainties, to classify the associated cases/issues, to predict the future results, and to explore possible solutions [29]. In wireless communication systems, problems that arise are frequently formulated as classification, detection, estimation, and optimization problems; for all of which ML techniques can provide elegant and practical solutions [25], [26] and [27].

2.4.1 Advantages of Machine Learning in MTC

As cellular system moves from one generation to the next, the number of reconfigurable system parameters also increases considerably. For instance, the number of configurable parameters in a 2G node is about 200, which increases to about 1000 in a 3G node, further rising to about 1500 in a 4G node. In ongoing 5G network, the number of system parameters is around 2000 [29], [32] and this number is expected to soar with each upgrade of the cellular system. Therefore, carrying out self-configuration, self-optimization, and self-healing operations will be tremendously challenging in 5G and beyond systems. To cope with these challenges and to handle these operations efficiently, ML techniques can play a vital role. Various issues such as link adaptation, resource allocation, user scheduling are usually too complex to be modelled, but usually they have hidden patterns, that can be explored using ML.

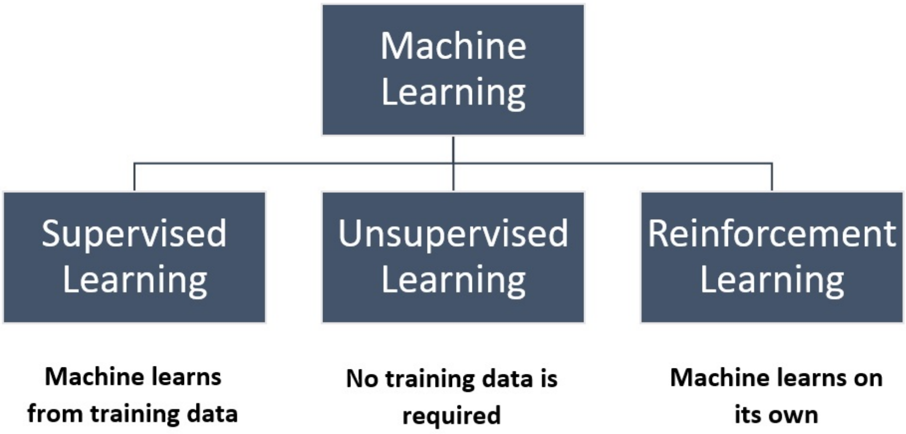

Figure 2.4: Types of Machine Learning

2.4.2 Types of Machine Learning

Machine learning is characteristically categorised into three broad classes [30], [31]: supervised learning, unsupervised learning and reinforcement learning. Fig.2.4 shows the types of different machine learning. In supervised learning, example inputs and their anticipated outputs are provided to the learning agent that targets to determine a general rule, mapping inputs to outputs. On the other hand, for unsupervised learning, instead of feeding prior input data to the learning agent, it turns to its own ability to find the embedded structure or pattern in its input, making it suitable for application in the AI category of cellular networks. Lastly, in reinforcement learning, the agent interacts with a dynamic environment in order to obtain its goal.

2.4.3 Reinforcement learning

Reinforcement Learning (RL) is a sub-field of machine learning, which has widespread applications in numerous areas such as medical diagnostics, psychology, neuroscience, informatics, cybernetics, control theory etc. RL differs from other learning techniques in terms of learning independency. In reinforcement learning, there are no data sets with correct answers as in supervised learning, but the reinforcement agent decides on the action to perform the given task. The reinforcement agent aims to maximise the cumulative reward in

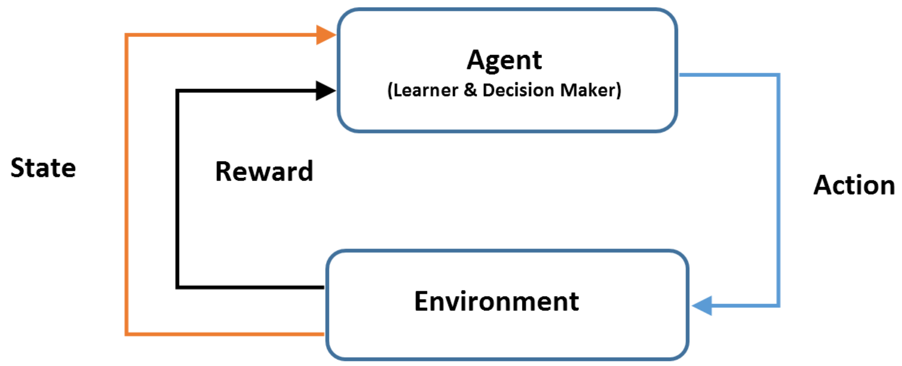

Figure 2.5: A basic block diagram of Reinforcement Learning (reproduced from [34])

a specific situation by applying a trial-and-error method from the direct interaction with the environment [28], [33]. There are four main sub-elements of RL system: a) a policy, b) a reward function, c) value function, and d) a model [34]. Fig.2.5 shows a basic diagram of RL.

- A policy is a fundamental part of RL that tells the agent what action to take at a particular state in the environment at a given time. This is also termed as a policy function. The interaction between the agent and its environment occurs in discrete time steps, where the agent selects an action purely based on a trial-and-error approach and changes its current state to the next state. Based on the outcome, the current policy is then formulated or reconstructed.

- At each step, an agent selects an action and the environment provides a reward based on the consequence of that action. The reward is a feedback, in the form of a scalar value, from the environment which helps to decide the next choice of action. The agent's objective is to maximise the total reward over the long run by altering its current policy according to its reward function.

- While the reward function indicates what is good in each step, a value function, on the other hand, indicates what may be the appropriate action in the long run. The value function helps to decide whether an action is desirable for long-term taking into account the total amount of reward an agent may expect to receive.

- Lastly, a model is something that represents the behavior of the environment by showing the correlation between the different RL elements including state, action, reward and transition probabilities between state, action and future state.

Model-based Reinforcement Learning:

The type of RL that requires a well-developed mathematical model which provides an accurate and comprehensive relationship between different elements, is termed as model-based RL. An example of model-based RL is Dynamic Programming (DP) [34] and is described in the following section.

Dynamic Programming (DP): Dynamic Programming (DP) is a model-based RL approach to achieve optimal policy or a solution. This method includes RL elements such as state, actions, rewards, and transition probability to construct the model of the environment as a finite Markov decision process (MDP). When an agent performs an action, the current state changes to the next and a probabilistic reward is received. Using dynamic programming algorithm, a policy is then computed to select the most appropriate action in the current state of the environment. After a series of policy iteration and policy improvements on current policy, an optimal policy is learnt in dynamic programming algorithm. DP algorithms are not widely used in solving RL problems, as they require an assumption of the perfect model of the learning environment as well as their great computational expenses [34], and therefore, is not considered in this thesis.

Model-free Reinforcement Learning:

Alternative to model-based RL techniques, model-free RL techniques do not require a model and are explicitly trial-and-error based. In a model-free RL technique, a policy is calculated directly from the experience gained through recursive interaction with the environment. Monte Carlo (MC) RL and the Temporal Difference (TD) learning are two types of model-free RL. In the following sections, we have discussed these two types of RL methods.

Monte Carlo: Monte Carlo methods require only experience which is sample sequences of states, actions, and rewards from interaction with an environment. Each experience is divided into episodes which eventually terminate no matter what actions are selected. Only after the completion of an episode, value functions and policies are updated. Thus Monte Carlo methods are considered to be incremental through an episode-by-episode sense rather than a step-by-step sense. We intend to use learning in this work in slot/preamble selection strategy, which will require a step-by-step incremental update of the learning process. Therefore, Monte Carlo is also not considered in this thesis.

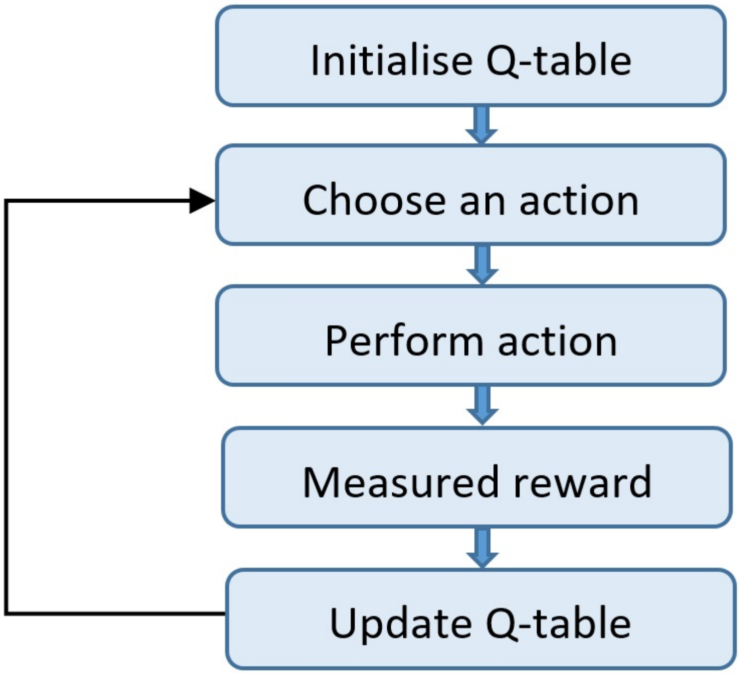

Figure 2.6: Q-Learning process

Temporal Difference (TD): Temporal Difference (TD) is a combination of DP and MC methods. TD methods learn from raw experience without any prior knowledge of the model of environment dynamics similar to MC method. Like DP, the agent in TD updates the value function after each step to learn a policy in order to select the next action to reach the target state. Q-learning is an example of off-policy TD method [88] where an agent recursively interacts with the environment purely through trial-and-error iterations to gather the information related to the environment and learn the most appropriate solution. These features of Q-learning TD methods makes it suitable for our application of slot/preamble selection and therefore, is considered in this thesis. In the next section, we provide an overview on Q-learning: off-policy TD method.

Q-Learning: Off-policy TD control

Q-learning is a model-free TD RL technique that enables the agent to learn an optimal policy from experience based on a trial-and-error approach in an unknown environment. This learning method does not require established state to action policies. The experience of an agent is represented by a Q-value which is a function of state-action pairs to learned values. By taking actions, an agent moves from one state to another. Each action offers the agent a reward and the agent targets to maximise the total reward. The agent achieves this using a table called the Q-table, where there are a tuple of states and a tuple of actions.

The agent always chooses the action that results in the highest Q-value. Q-learning is also called off-policy TD method because the target-policy is different from that of the behaviour policy, where behaviour policy determines how an agent will behave in a given state, on the other hand, target policy is concerned with updating its estimate of Q-values using rewards it receives. Fig.2.6 shows basic Q-learning process and the detailed Q-learning implementation is presented in chapter 7.

2.5 Summary

This chapter has provided background information related to the research presented in this thesis. A general overview of M2M communication is given with its various applications and requirements in section 2.2. The challenges and advantages of cellular networks for M2M communication are discussed, and some potential enablers for M2M communication are also provided in section 2.3. Advanced Random Access scheme is one of these potential enablers and a detailed overview on Random Access Channel in cellular network is later provided in chapter 3. Additionally, an advanced two-step Random Access scheme is discussed in Chapter 5. In section 2.4, as an emerging solution, machine learning (ML) techniques with its types and advantages are discussed. The concept of Reinforcement Learning (RL) has been introduced, where three different solution methods for RL are briefly described and the reason for choosing the Q-Learning algorithm in this research is explained in section 2.4.3. A more detailed overview on Q-learning is provided in chapter 7.

Chapter 3

Random Access Channel Management in Cellular Network

3.1 Introduction

The objective of this chapter is to provide background information on Random Access Channel (RACH) Management and its impact on MTC. In section 3.2, a detailed overview of RACH in LTE is discussed with a brief discussion on RACH limitations due to MTC. Section 3.3 outlines a detailed discussion on RACH overload control mechanisms including both 3GPP and non-3GPP solutions and finally, we summarise the chapter in section 3.4.

3.2 Random Access Channel in LTE

Long Term Evolution (LTE) is a standard for the wireless network and is an evolution from the standard Global System for Mobile communication (GSM) and High Speed Packet Access (HSPA) established by the 3rd Generation Partnership Project (3GPP). Some features of LTE include ultra-broadband internet service, fast data rates (100 Mbps – 1.0Gbps), high-speed handoff procedure, Multiple Input Multiple Output (MIMO) technology, worldwide roaming service, etc. [42]. In Long Term Evolution (LTE), devices need to perform an initial attachment procedure called Random Access process, using Random Access Channel prior to data transmission. However, LTE networks are designed mainly for H2H, so when a massive number of MTC devices perform initial access request procedures at the same time using RACH with a limited resource, congestion occurs, which causes a network overload scenario. To mitigate these challenges, in this work LTE RACH scheme is considered as a baseline in order to establish improved RACH protocols so that it can cope with a massive number of M2M devices in next-generation communication paradigms. In the following subsections, we will first demonstrate the uplink resource structure of the LTE networks and then elaborate on the RA process.

3.2.1 Uplink Resource Structure of the LTE Network

In LTE networks, data packets and signalling messages have to be delivered in different time-frequency resources. In the time domain, the duration of one radio frame is 10ms, where each frame is identified by a number known as system frame number (SFN). One frame is divided into 10 subframes of 1ms each and each subframe is again divided into two equally sized slots of 0.5ms, resulting in 20 slots per frame. Each slot contains either six or seven OFDM symbols, depending on the Cyclic Prefix (CP) length, where a symbol is the smallest unit in a resource block. A resource element represented is the smallest modulation structure in LTE, which is equal to one 15 kHz subcarrier in the frequency domain. Resource elements are aggregated into Resource Blocks (RB), which is the minimum allocation unit, where twelve consecutive subcarriers together make an RB with a bandwidth of 180 kHz in frequency and 6 to 7 symbols in the time domain. Therefore, as shown in Fig 3.1, a single RB is allocated in a slot period and consists of 12 subcarriers in the frequency domain and 7 to 6 symbols (depending on the type of the cyclic prefix used) in the time domain [43].

LTE Uplink Channel Mapping: In LTE, different types of channels are used to transport data across the radio access network where channels are separated based on the types of information they carry and process. In LTE channels are divided into three categories and these three types of channel are present in Downlink as well as Uplink direction [41]. In our work, we are focusing on Uplink Channels only however, the details of LTE channels can be found in [43], [44] and [45]. Fig.3.2 shows LTE Uplink Channel mapping where LTE Uplink Channel are divided into three categories.

Uplink Logical Channel:

- Common Control CHannel (CCCH), which is common to multiple UEs.
- Dedicated Control CHannel (DCCH), which is used to transmit dedicated control information for a particular UE.
- Dedicated Traffic CHannel (DTCH), which is dedicated traffic channel for a particular UE.

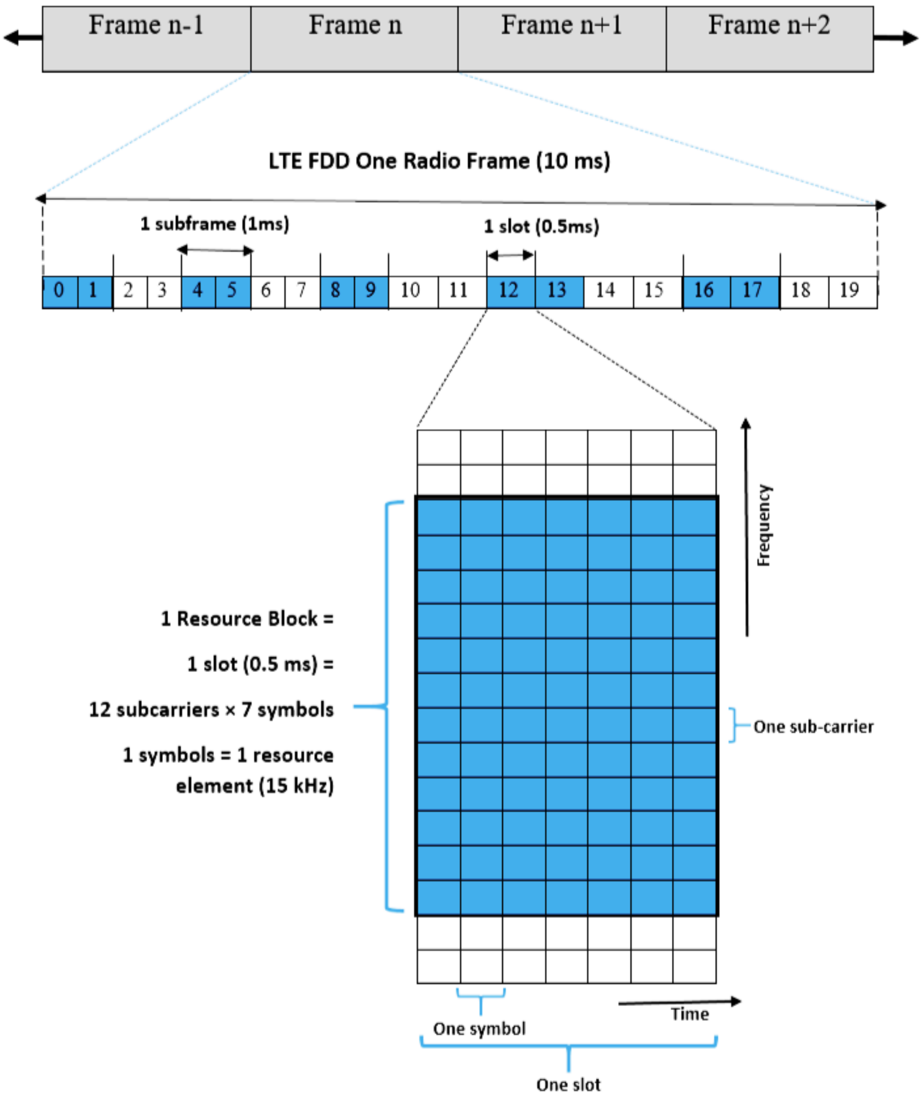

Figure 3.1: Time-frequency resource structure of the LTE system.

Uplink Transport Channel:

- Uplink Shared CHannel (UL-SCH), which is the main channel for uplink data transfer.
- Random Access CHannel (RACH), which is used for random access procedure.

Uplink Physical Channel:

- Physical Uplink Control CHannel (PUCCH), which delivers control signalling, such as channel quality indicator report.
- Physical Random Access CHannel (PRACH), which is used for transmission of random access preambles.
- Physical Uplink Shared CHannel (PUSCH), which is used to carry user data packets and radio resource control messages.

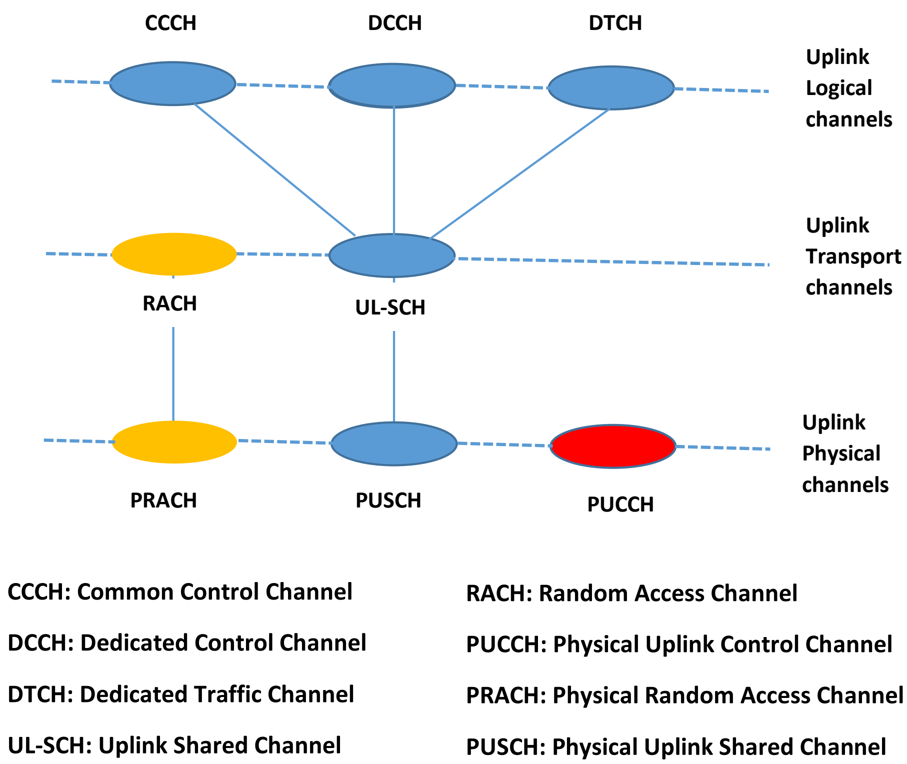

Figure 3.2: LTE Uplink Channel mapping

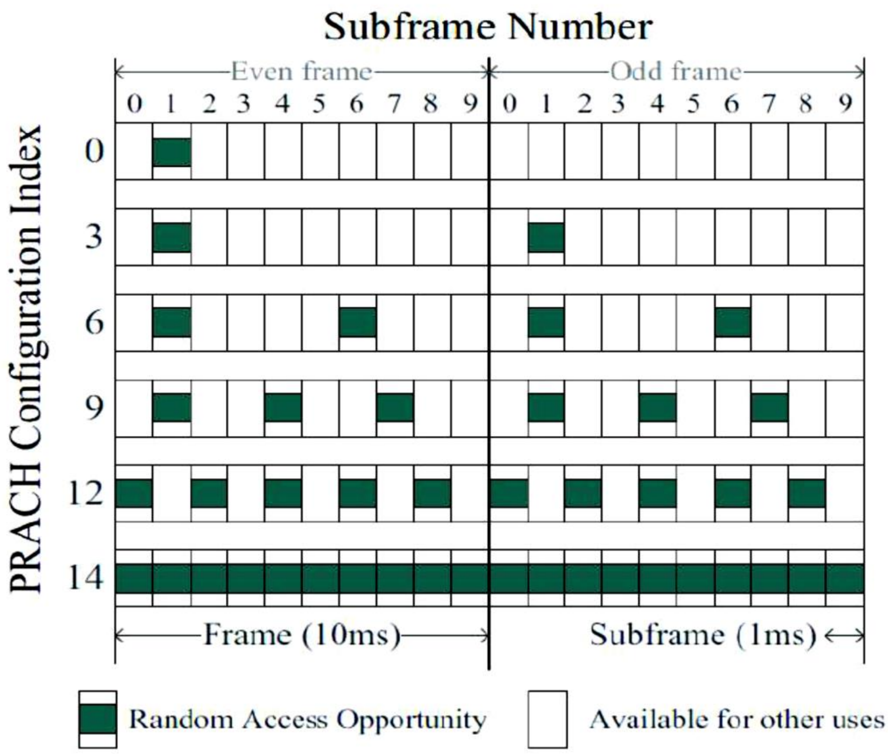

Figure 3.3: Representation of PRACH configuration index (directly reproduced from [17])

PRACH Configuration: In Fig.3.3, it is shown that each System frame (SF) contains Random access attempts called Random Access Opportunities (RAO) that can only be transmitted in specific sub-frames, which are mentioned as random access slots (RA slots). One or more RA slots may be supported in each frame, and the number of RA-slots depends on the PRACH configuration index, and this will be allocated periodically and the period is broadcast by the eNB. LTE defines up to 64 possible configurations [46] that vary between a minimum of 1 RA-slots in every 2 frames to a maximum of 1 RA slots per 1 subframe, i.e., every 1ms. In Figure-3.3, some PRACH configuration indexes are presented where filled squares represent RA slots.

From Fig.3.3 based on different PRACH configuration index values, it is clear that RACH requests are restricted to RA-slots. Due to this nature of the PRACH configuration arrangement, LTE adopts the s-ALOHA protocol to control Random Access Procedure. Random Access Procedure is described in the next section.

3.2.2 RA Procedures in LTE

This section describes the random access scheme used by the conventional LTE network. As mentioned earlier, the Random Access Channel (RACH) is the initial access through which a user is connected with the network. A device (H2H and M2M) must initiate the access procedure to establish a connection to the BS/ eNB/ access point in the following situations [17], [47]:

- To establish initial access.
- To establish uplink synchronisation.
- To perform seamless handover (change of eNodeB).
- To re-connect to the network in case of radio link failure.

In order to handle all these situations, random-access procedures in LTE-based cellular systems can be categorised into two different forms:

- Contention based (Supports collision)
- Contention-free (No collision and applicable to handover only)

Each random-access procedure of 3GPP LTE consists of the following four steps as shown in Fig.3.4:

Random-access Preamble transmission: In this step, a UE sends its access request by transmitting one out of the available preamble sequences via Msg-1. The preamble is selected in a random manner and carried in the Physical Random Access Channel (PRACH), which is a part of an uplink resource of an LTE network. A Random Access Response (RAR) window is set up to wait for the RAR. If a UE does not receive the RAR in a RAR window, it means the initial access has failed and UE shall randomly back-off for a period between 0 to a back-off indicator parameter (BI) value

Random-access Response: In this step, the eNB transmits the access response to the detected preamble sequence by sending an RA Response (RAR) via Msg-2 on the Physical Downlink Shared Channel (PDSCH).

Scheduled Transmission: After receiving the RAR at step-2, the UE transmits a connection request such as Radio Resource Control (RRC) connection request followed by Msg-3 in order to establish a connection using Physical Uplink Shared Channel (PUSCH).

Contention Resolution: In this step, when eNB receives Msg-3 it replies Msg-4 to confirm that the connection is successfully established and the status changes to RRC_CONNECTED.

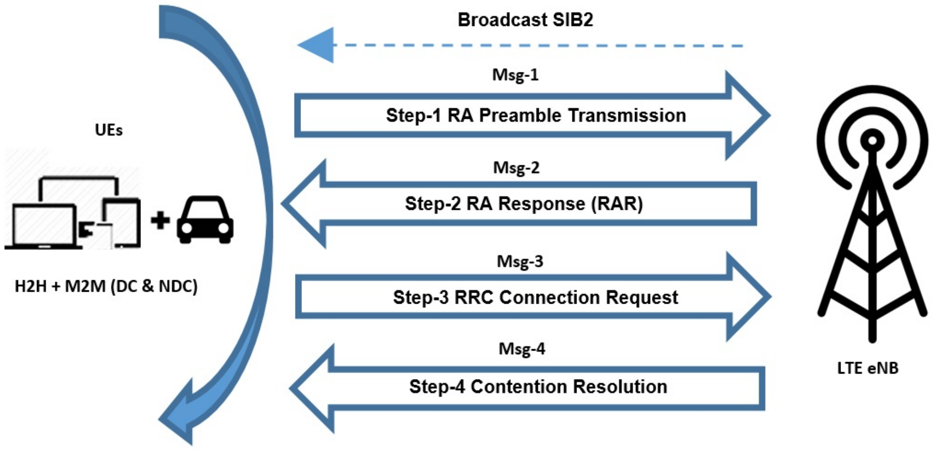

Figure 3.4: Random-Access Procedure

Otherwise, if the Msg-4 is not received by the UE, the RA is declared as failed and the UE needs to restart the RA process all over again until the allowed preamble retransmissions are reached. Further details on the RACH procedure in LTE can be found in [48].

As shown in Fig.3.4, the eNB broadcast System Information Block2 (SIB2) message where all the information related to the RACH procedure is carried out. The details about the information elements in the SIB2 message can be found in [19]. As part of the information, the "PRACH Configuration Index" ($PrachConfigIndex$) defines the UE when the UE itself is supposed to transmit the RACH request. The number of Random Access Opportunities (RAO) in an LTE cell depends on the $PrachConfigIndex$ and the number of preambles available for contention-based RA. In each LTE cell, there are up to 64 orthogonal preambles available created by the Zadoff-Chu sequence [49]. Out of 64 preambles, 54 are used for contention-based access, while the remaining 10 are dedicated for contention-free access [50]. Since the number of available resources is limited in comparison to the large number of access requests, if two or more devices choose the same preamble to send the connection request at the same time, a collision takes place and this kind of PRACH collision is called the "Contention" and the RACH process that allows this type of "Contention" is called "Contention based" RACH Process. In the contention based RACH process to solve the contention issue, eNB needs to go through an additional process called "Contention Resolution" step. Now let's assume that a collision happened at step-1, where two UEs select same preamble to sent Msg-1. In this case, both of the UEs will receive the same RA response via Msg-2 and both UEs will send Msg-3 through the same resource allocation to eNB. As a result, eNB will not be able to decode either of them and none of the UEs would have any response (Msg-4) from eNB and they think that the RACH process has failed and they have to start from step-1 again. The other possibility would be that eNB could successfully decode the message from only one UE and failed to decode it from the other UE. In this case, one of the colliding devices delivers the RRC connection and Contention Resolution message, while the other UE needs to wait for a random time before attempting a new request. Due to the limitation in the number of available resources compared to the massive number of access requests to be supported, contention-based RACH is considered in our work.

3.2.3 Conventional RACH Limitations

As the number of M2M devices has been growing rapidly, the load on the random access channel is also increasing. Consequently, devices attempting RA within a small time interval is causing an access problem referred to as the "massive access problem" as mentioned by 3GPP [35]. The worst-case scenario according to 3GPP is that thousands of devices may attempt to perform RACH within a 10ms time window [35]. However, at a peak traffic load where the number of access requests is maximum, the standard LTE random access mechanism suffers congestion due to the high probability of collision and this causes excessive time delay. In [51], the results showed that using standard current LTE medium access control system, the access delay may be intolerable when the number of devices exceeds 30,000 per cell. As a result, for some delay critical M2M applications that require ultra-low latency (e.g., e-health, intelligent transportation system), the standard LTE will not be adequate, which may cause a sharp degradation of QoS.

One possible way to mitigate the overload problem is to increase the number of RA opportunities per frame, but this causes a fall in the existing amount of resources for data transmission and therefore, it reduces the data transport capacity of the uplink channel. To diminish this problem besides improving the legacy system, it is significant to provide an effective approach for managing the massive access in the radio access network to reduce the network overload as well as to minimise the latency.

3.3 RACH Congestion Control

In this section, we provide a comprehensive review of RA congestion solution proposals to control the RACH overload problem caused by M2M traffic in LTE systems. The proposals are categorised under two classes: 3GPP and non-3GPP specified solutions.

3.3.1 3GPP Specified Solutions

In [52], 3GPP has proposed six basic mechanisms for RA overload control and in the following, we briefly describe the principle of these techniques.

Access Class Barring (ACB): ACB is a renowned mechanism in controlling RA congestion by decreasing the access arrival rate. ACB can define 16 access classes [53] and [54], each class operates on two factors: a set of barring access classes (ACs) in which devices are classified, and a barring time duration (T_b). First, the eNB broadcasts the ACB parameter, p (ranging from 0 to 1) to the MTC device and each MTC device also generates a random number, r (between 0 and 1) uniformly. If $r<p$, the device is permitted to transmit their RA preamble, otherwise the access is barred and the device has to wait for a random backoff time based on the barring time duration (T_b). However, in a peak congestion condition when a massive number of devices try to connect in a very short time, the value of p might be set to a very low value which leads to intolerable delay. In [55], a Dynamic ACB scheme is approached and a Prioritised RA jointly with dynamic ACB is proposed in [56] to improve the performance of the RACH channel.

MTC-Specific Backoff: In this scheme, when a device faces a collision, it waits for a fixed back-off time and then retransmits the connection request. Although the network performance under low congestion levels increases using this scheme, in high-congestion levels the network performance is reduced [57]. In [58], a separate back-off scheme is suggested for separate user groups: delay-sensitive M2M with H2H as group 1, and on the other hand, delay-insensitive M2M as group 2. Devices in group 2 have a longer backoff time compared to group 1. Although this scheme can provide some enhancement for low congestion [59], it is not satisfactory to handle peak congestion levels.

Dynamic Resource allocation: In this scheme, the BS allocates additional RACH resources dynamically in the time domain or frequency domain or both by predicting the congestion level of the access network overload caused by MTC devices [35] and [60]. In [59], a simulation result is presented by 3GPP showing that additional allocation can solve most of the congestion problem. However, allocating more resources for RACH will reduce the available resources for data traffic, which in turn causes problems in the network performance.

Slotted Random access: In the slotted aloha method, each MTC device is provided with a dedicated RA opportunity using only the slot allocated to the device [61]. These slots comprise an RA cycle, where eNB periodically broadcast the parameters of the RA cycle and access slots. For a large number of MTC devices the RA cycle is likely to be very large and as a result MTC devices might experience long access delay. Also each RA slot consists of 64 RA opportunities and in case of massive numbers of access requests there is a strong possibility that all 64 access attempts are made within a single RA slot, which will cause high collision rate in a slotted aloha based random access system. Moreover, in slotted RA scheme, while there could be very high load in some slots, some other slots may remain underutilized.

Separate Resource Allocation: In this approach, The MTC and HTC devices are delivered with different RACHs in an attempt to avoid the effect of RA congestion on HTC devices. The separation can be possible by assigning different RA slots for MTC and HTC devices or by splitting the available preambles into MTC and HTC subsets [60]. This separation technique might help to drop the negative impact on non-M2M devices.

Pull-based RA: All the schemes mentioned in earlier sections are categorised as push-based approaches in which RA attempts are random and are started by devices autonomously. On the other hand, in a pull-based method [52], the RA procedure is started by eNB. Therefore, the eNB can control the number of requests and mitigate congestion problems. The devices perform RA attempts only after getting paging messages from the eNB. However, the scheme needs additional control channel resources to page a massive number of devices. To reduce the number of paging loads, a number of MTC devices can be paged together by following a group paging method, an analytical model is developed in [62] for performance evaluation of group paging in LTE.

3.3.2 Non-3GPP RACH solutions for supporting M2M

In addition to the solutions specified by 3GPP, several academic, industrial, and governmental institutions have also proposed various RA congestion solutions to support the huge MTC in LTE networks. Some proposals are discussed in the subsequent paragraphs.

Group-based RA Scheme: A group-based RA mechanism is an addition to the pull-based group RA model. Based on some specific criteria such as similar QoS/delay requirements, MTC devices can be grouped in a particular region and the RA procedure can be assigned on a group basis in order to minimise the network congestion. In [63], a two-layer device segregating technique to reduce congestion is proposed, where in the first layer, devices are grouped into several paging groups. Devices within a paging group are then partitioned into different access groups. For each group, a group head is assigned who is responsible for communicating with the eNB. Another group-based approach is proposed in [64], where dividing the cell coverage area into different spatial groups is mentioned. The idea behind the cell division approach is to permit the use of the same preambles in the same RA slot by different groups of MTC users if the distance is not larger than the multi-path delay spread. In [65], a cluster-based approach is proposed for mitigating the inefficiencies of the ACB algorithm. In [66] a technique is proposed based on groups in which M2M devices are grouped according to their characteristics (access speed) and requirements (maximum tolerable delay). Hence a decision-making step is taken upon reception of the data about the characteristics after the third step of the RACH procedure in LTE.

Code-Expanded RA Scheme: In this scheme, a codeword (set of preambles) is transmitted to execute an RA process instead of a single preamble. A virtual RA frame is considered which contains a group of RA slots, or a set of preambles in each slot. This allows expansion of the number of contention resources, and reduction in the number of collisions [67].

Self-optimisation overload control RA: This self-optimising mechanism is proposed in [18], which configures the RA resources depending on the load condition. It encompasses a combination of other solutions, specifically Separation of RACH Resources, ACB Schemes, and Slotted-Access scheme. The LTE-A ACB scheme is modified by adding two classes of M2M devices, i.e. high priority and low priority. The ACB scheme is applied to the next attempt when a device is not granted access in the first attempt. After receiving an RAR, a device sends the number of retransmitted preambles to the eNB within message 3, which is used for overload monitoring and adjusting the RA resources according to the congestion level of the RACH. If the RA slot usage reaches the maximum accessible limit, the lowest priority M2M class devices are temporarily restricted from accessing the network until overload conditions recover.

Prioritised Random Access: In this scheme, applications are divided into five categories: HTC, high-priority MTC, low-priority MTC, scheduled MTC, and emergency services. Also, virtual separation of the RACH channel is applied into three classes i.e. HTC, random MTC, and scheduled MTC (emergency services) [56]. A prioritisation is accomplished by applying distinct back-off window sizes to guarantee QoS using a prioritised access algorithm based on the mentioned classes. It has been reported that this type of scheme is better than other EAB methods in terms of access delay and probability of success, but it still needs the prohibition of an M2M device for an amount of time [18].

In the study above, we have discussed existing methods for controlling the RA procedure in corporations with M2M from 3GPP and non-3GPP perspective. We have highlighted some key issues related to the enormous number of devices trying to connect to the network at the same time. It was noticed that some existing RA congestion control methods are not appropriate for solving the RACH overload problem in massive M2M scenarios [68]. ACB based methods are targeted to lessen collisions, preamble-splitting methods perform decently to protect the H2H QoS, but both schemes result in intolerable delay. Resource allocation methods are suitable for both counts; however, like back-off-based schemes, it decreases the general throughput. In contrast to the ACB and back-off schemes [69], the slotted access approach is advantageous in several ways such as offering better access rates, in terms of complexity and has minimum influence on H2H traffic. The simulation results presented in [69], demonstrate that the slotted access scheme outperforms others in the case of access rate even for randomly assigned slots. The slotted access uses the already provided paging mechanism, whereas ACB imposes signalling overloads resulting in added complexity. [70] shows a comparison of ACB, back-off, and slotted access approaches for overcoming the LTE RACH overload problem, summarising the minimal effect of the slotted access method on H2H access traffic in contrast to the other methods. Nevertheless, some research studies show that the combination of two or more methods could result in better performance aimed at overload controlling.

3.3.3 Emerging Solutions for MTC

In the following, we have provided some of the emerging research directions to address the RAN congestion problem in a massive MTC scenario.

Learning-Based Approaches: Resolving the RAN congestion problem in cellular networks discussed in the previous section has been a major concern in recent times. As an attempt to mitigate this problem, a Reinforcement Learning (RL) scheme has been applied in [71] for selecting an appropriate BS for the MTC devices in order to avoid access network congestion and minimise the packet delay. Also, a Q-learning-based scheme was studied in [72] with the objective of supporting MTC traffic in the present-day cellular networks where the MTC devices learn to avoid collisions among each other without assistance from a central entity. As a result of the learning convergence, each MTC device acquires a unique RACH slot. Additionally, in [73], another Q-learning-based unsupervised algorithm was studied in order to select an appropriate BS for MTC devices taking into consideration different QoS parameters in dynamic network traffic conditions. Furthermore, in another study, [74] a hierarchical stochastic learning algorithm was applied for enabling each device to make access decisions based on common control information broadcasted from the BS. Likewise, in [75], another Q-learning algorithm was applied in the fashion that a barring factor can be allocated dynamically to the MTC device in the ACB scheme.

Grant-free random access: A grant-free (GF) random access scheme has been proposed for MTC in the 5th generation (5G) by the 3GPP [89]. GF random access allows active UEs to transmit their data to the base station (BS) directly where the traditional 4-step handshake-based grant acquisition phase is skipped [90]. The motivation of GF is to reduce the access latency and control signaling overhead with a two-step RACH scheme compared to the conventional 4-step RACH [91]. However, one major drawback is the collisions that occur when multiple MTC devices select the same uplink resources. Another challenge is the synchronisation among the MTC devices due to the omission of RA process [92]. In our work, we adopted a 2-step random access process, where we modified the traditional 4-step RACH to a 2-step RACH providing priority to a specific user groups called delay-critical UEs. Our approach is based on LTE-RACH, it is applicable to 4G and beyond whereas GF random access is suitable for 5G new radio (NR). The details of 2-step RACH is given in section 5.3.

3.4 Summary

This chapter presents an overview of RACH in the LTE cellular network. Firstly, in section 3.2.1, the Uplink Resource Structure of the LTE Network is introduced in order to understand the RA mechanism in the conventional cellular network. Secondly, Random-Access procedure is introduced and the conventional 4-step RACH is described. In this work, we later developed a RACH simulator using Riverbed Modeller and the design methodology is provided in chapter 4. Next, the limitations of conventional RACH due to the massive number of MTC devices is described in section 3.2.3. A literature review on RACH overload control mechanism is given in detail including both 3GPP and non-3GPP based solutions in section 3.3. As the literature review indicates, a combination of two or more overload control methods could result in better performance in solving the RACH congestion issue and to support massive machine-type communications. In this work, we have considered this hybrid approach to solve the RACH congestion issue. We have introduced a novel two-step RACH scheme using slotted-ALOHA (SA) and Group-based RA in Chapter 5 and the main idea behind this scheme is minimising the signalling overhead compared to 4-steps conventional RACH. Additionally, we have combined Dynamic Resource Allocation, Group-based RA and Prioritised RA schemes as a hybrid RACH mechanism in order to provide dynamic prioritisation for specific user groups in chapter 6.

Chapter 4

Modelling and Performance Evaluation Method

4.1 Introduction

In this chapter, introduction to the methods used to develop MAC layer simulation models for an uplink cellular system (random access channel) using a software simulation tool is provided. In section 4.2, a brief overview of system simulation on communication and the types of simulation software used in this work is delivered. A reference model of the developed simulator and simulator design steps are described in section 4.3. Section 4.4 describes the performance metrics that are used to evaluate network performance. Section 4.5 introduces the traffic models and finally, the chapter is concluded in section 4.6.

4.2 Simulation in Communication System

In communication, a system has to go through a number of steps throughout the development process, whether it is a completely new design or an alteration of an existing system. Although the design process determines the final behaviour of a system, validating it through design evaluation is crucial. The techniques used in a communication system for performance validation are as follows:

- Analytical Models
- Experimental measurements
- Simulation Technique

In the early design stages, analytical models are often incorporated as a performance validation method based on a simplified model. The Erlang-B formula for call blocking probability is a well-known example, which is used for circuit-switched networks [76]. On the other hand, a communication system may be validated by creating hardware prototypes and applying measurement methods. However, most of the time this method is avoided since it can be very costly and inflexible in nature. In contrast to these techniques, simulation-based validation methods are more flexible, comparatively inexpensive and can also be applied effectively in complex systems [77]. Simulation is a powerful tool to gain an in-depth overview of different aspects of a computer network technology, however, there remain some limitations to it for example credibility and validation, scalability limits [97]. Instead of these disadvantages, this research work opts toward using simulation-based methods for validation because it is reasonable compared to the other methods.

4.3 Simulation Software Tools

To simulate a communication system, one of the most important steps is to choose a software tool according to the complexity of the desired task, which means different layers of a communication system require different types of simulation tools. This work has utilised the tool called Riverbed Modeller, previously known as OPNET. Using the Riverbed Modeller, a user can either choose the customised simulator or can develop the algorithm according to his preferences.

4.3.1 Riverbed Modeller

Riverbed Modeller is a simulation tool with a virtual environment for modelling, which can be used to simulate wired and wireless network technologies. It is an event-driven and interrupts based simulation tool which allows a user to develop advanced protocols and is capable of simulating a close-to-real communication environment.

Fig.4.1 shows a reference model of the developed simulator using Riverbed Modeller, which is used in this work. The structure of the developed simulator can be divided into three levels: network model, node model and process model. The network model includes physical and logical part of a network model such as nodes, subnets and their mobility. In this work, a coexistence of H2H and M2M device is considered in the LTE network scenario. The network model has three main components H2H node, M2M node and eNB Node (base station). The node model defines the inner structure of nodes and base stations. Due to the characteristic difference between H2H and M2M, different traffic model is considered for

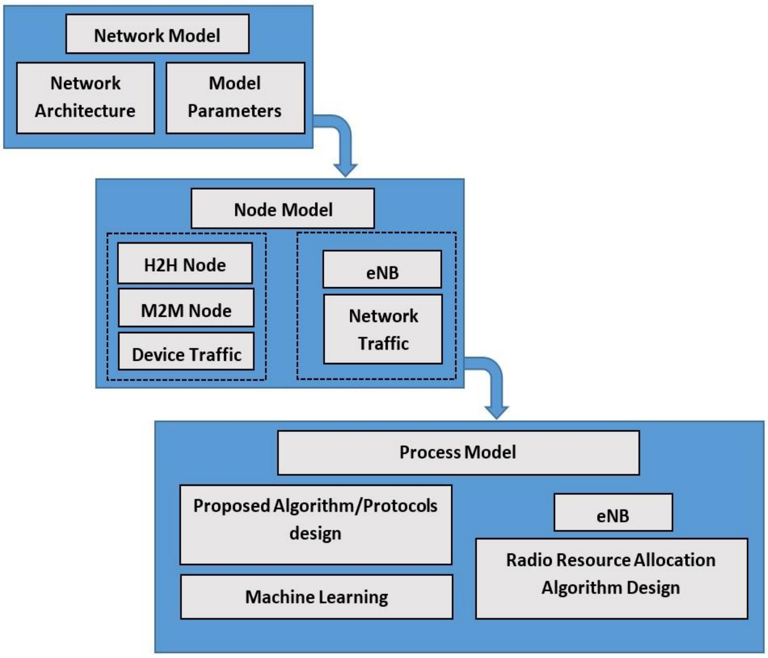

Figure 4.1: A reference model of the developed simulator in Riverbed Modeller

both nodes, also each group have their own parameter values such as packet generation time, arrival time, transmission time. The process model is developed with a computer language called PROTO- C, which allows a user to model a wide range of system and provide a flexible programming ability. Due to this opportunity, in this work different algorithm is developed. In eNB node, protocols for radio resource allocation such as dynamic prioritisation, coded preamble based collision model for the proposed RACH mechanism are designed. Also in the device (H2H/M2M) node, protocols such as machine learning algorithm are developed using PROTO-C in the process model.

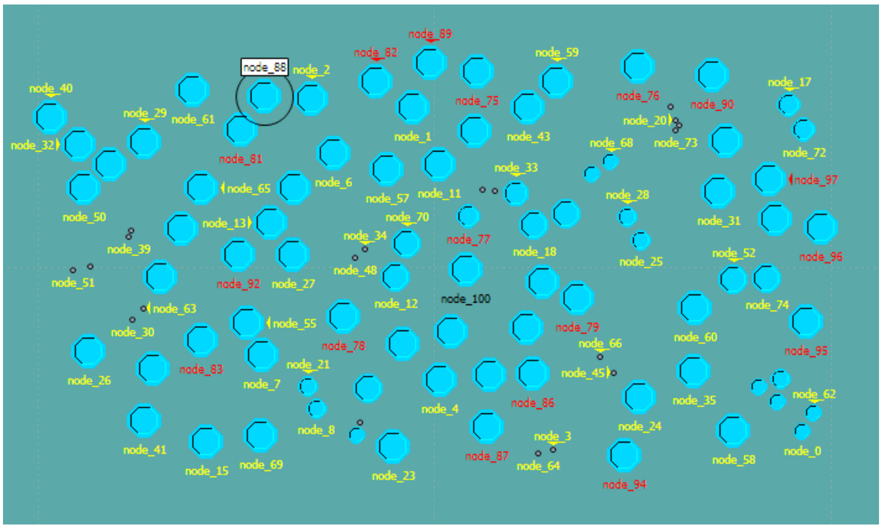

Figure 4.2: The Network Domain

4.3.2 Simulator design in Riverbed Modeller

In our work, we have considered Riverbed Modeller to design and develop our simulator where the structure of the developed simulator is divided into three parts: the network domain, the node domain, and the process domain. Each domain uses a single paradigm to perform corresponding tasks to the modelling process. The following section gives an overview of each domain in detail.

Network Domain: The Network Domain (shown in Fig.4.2) consists of objects representing physical or logical parts of a network model that mainly includes three types of components: subnets, communication nodes, and communication pipelines (invisible for wireless networks) [93]. In a network domain, nodes/subnets can be fixed or mobile. In our work, we have considered a large number of fixed nodes and used a tool called the Rapid Configuration Tool for creating the network topologies. There are three different types of node included in the network domain namely: Human type nodes (H2H), Machine type nodes (M2M) and Base stations (eNB). M2M nodes are further categorised into two types: Delay-critical (DC) M2M and Non-delay-critical (NDC) M2M.

Node Domain: The second domain is called the node domain which defines the inner structure of each of the nodes of the network domain. It includes several modules with specific functions: processing, transmission and reception. Fig.4.3 shows UE and eNB node, these include five modules: a transmitting processor module (RACH-MAC), a receiving processor module (Receiver), a packet generator module (GEN), a wireless transmitter

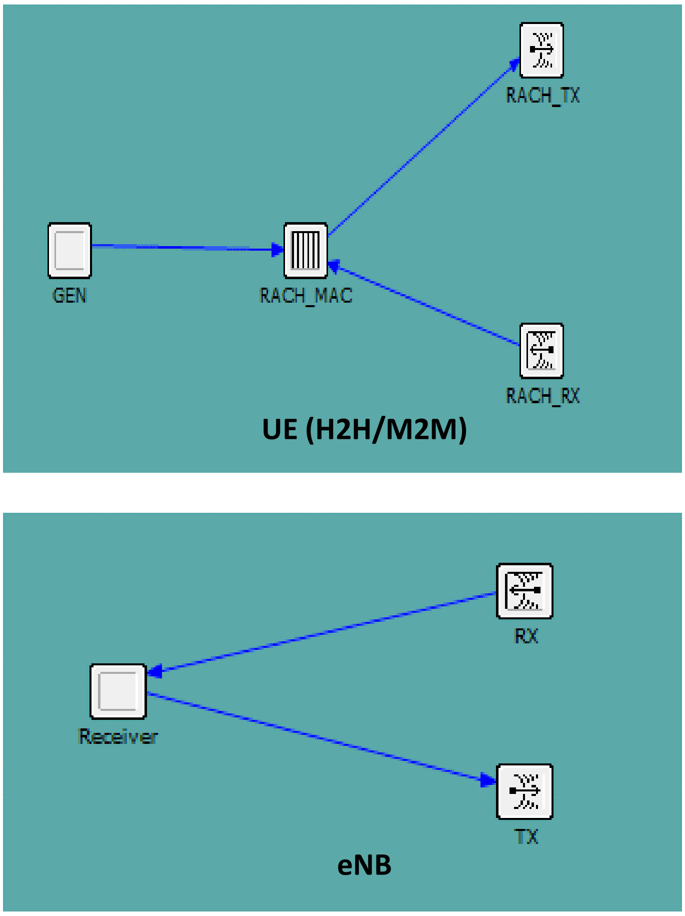

Figure 4.3: The Node Domain

module (TX), and a wireless receiver module (RX). In our work, the UE (H2H or M2M) nodes have four modules. The packet generator module (GEN) works as a random access request generator as a packet. The transmitting processor module (RACH-MAC) works as a RACH access control processor which can queue packets in its memory and schedule their transmission/retransmission. The wireless transmitter module (TX) sends RACH requests as packets. In the eNB node, wireless receiver module (RX) receives the RACH request and forwards it to the receiving processor module (Receiver) that can be used to process the incoming requests/packets and collect the final statistics. Finally, eNB sends random access response using wireless transmitter (TX).

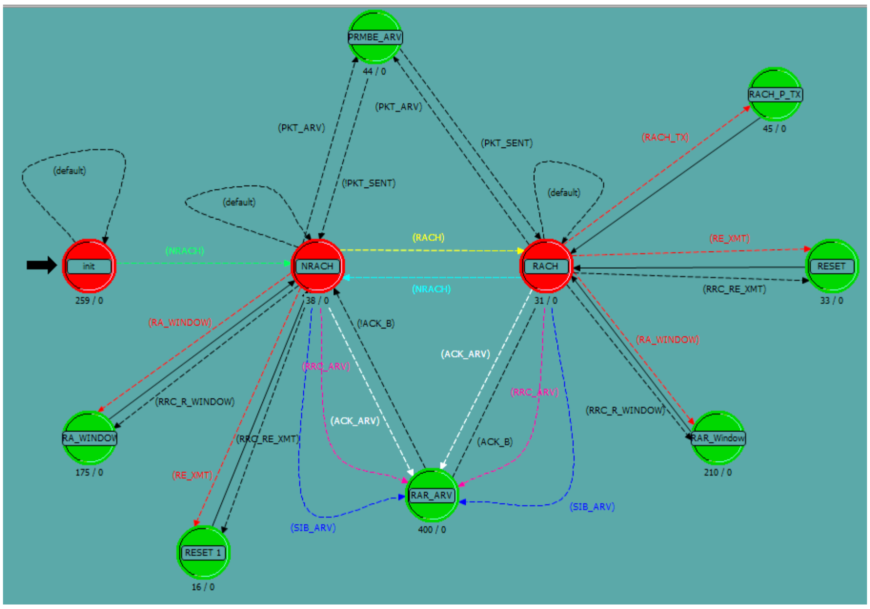

Figure 4.4: The Process Domain

Process Domain: In the process domain, the detailed protocol mechanisms are designed and implemented. It is the lowest level of system design hierarchy [94]. Fig.4.4 shows a typical process domain. The language used to develop the protocol is called PROTO-C [95] which provides a flexible programming ability to model a wide range of systems. Each process model usually consists of several states and state-transitions which represent a specific activity the node may perform. Only one state can be active at any particular instant and during its activation period, the corresponding codes are executed by the system until the end of current state. Each state are divided into two parts: the enter executive and the exit executive. The transition from one state to another can be immediate or depend on a condition defined by the user. There are two types of states: forced state (shown by green colour in Fig.4.4) and unforced states (shown by red colour in Fig.4.4). In the forced states, the module directly runs the exit executive after completing the enter executive. On the other hand, in the unforced states, the execution is paused after the entry code is executed and waits for the next interrupt where the type of interrupt and its condition determine the next state. In this work, we have considered different types of states and their corresponding conditions to design our proposed protocol. For example, when an UE enters the RACH-TX state (shown in Fig.4.4) it will transmit a request.

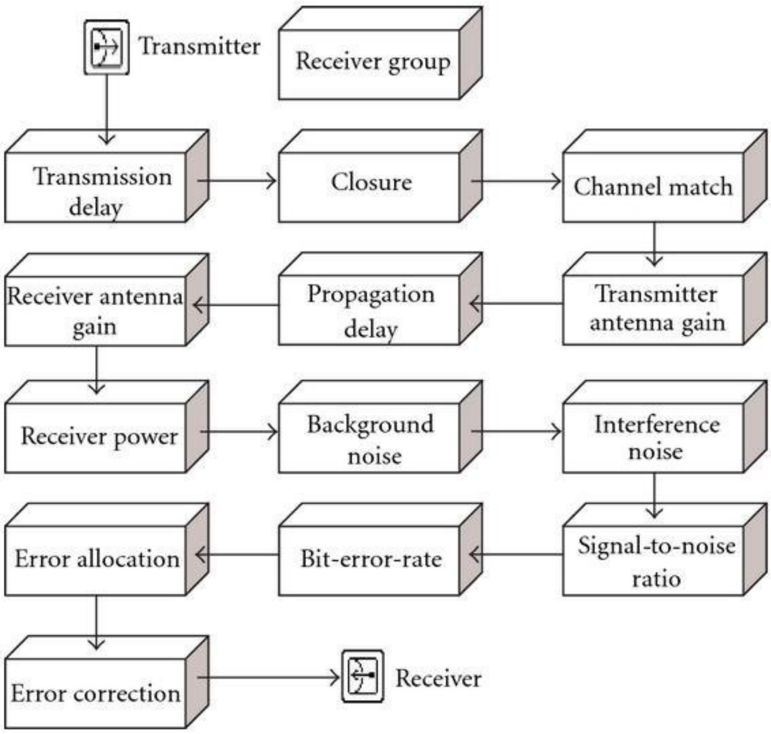

Figure 4.5: Riverbed Modeller Radio Transceiver Pipeline Stages, reproduced from [96]

4.3.3 Radio Receiver Pipeline

Riverbed Modeller offers open and modular architecture for implementing link behaviour called the Radio Receiver Pipeline. The role of Radio Receiver Pipeline is to simulate the basic influence of all elements (such as channel fading, noise, interference, transceiver hardware) in the network on transmission. These include different physical phenomena as well as characteristics of the physical layer (such as channel and error correction) so that differences in behaviour and timing of the radio link can be taken into account for a given transmission. Each transfer of packet goes through a series of computational procedures forming a complete radio transceiver pipeline containing 14-procedures. The sequencing and interfaces of these 14-procedures are standardised for each type of link and this way, the Radio Receiver Pipeline is able to implement link behaviour. Fig.4.5 shows the radio transceiver pipeline stages of Riverbed Modeller. The failure of certain stages will change the rest of the stages and considers the failure of the transmission. All the stages have default configurations but they can be modified by the user to satisfy different requirements.

Propagation Delay: In this work, propagation delay is calculated using pipeline stage-5 which is Propagation delay (shown in Fig.4.5). The purpose of this stage is to calculate the amount of time required for the packet's signal to travel from the radio transmitter to the radio receiver. This result is generally dependent on the distance between the source and the destination. The Kernel uses this result to schedule a beginning-of-reception event for the receiver channel that the packet is destined for. In addition, the propagation delay result is used in conjunction with the result of the transmission delay stage to compute the time at which the packet completes reception.

Error correction: In our work, we have not considered propagation loss and we have only assumed that one packet will be dropped if it suffered one or more collisions. To determine whether the arriving packet can be accepted and forwarded to the receiver's neighbouring modules in the destination node or not decides by the final pipeline stage called error correction (ecc) stage.

4.4 Performance metrics

The performance of a system is estimated through the evaluation of some performance metrics. Some of them have been used in this research work and these will be explained in this section.

4.4.1 Offered Traffic

According to Harada and Prasad, offered traffic is the total number of active packets that include new packets generated as well as retransmission packets within the given time interval [78]. In this work, both new packet and re-transmission packets are considered as this work is concerned with a signalling channel carrying bursts of information comprising multiple packets, where retransmission takes place when an access request fails because of collision. In this thesis, the total number of RACH requests within a given time interval is defined as the offered traffic. Offered traffic can be measured in a unit called the Erlang (E) [79], which defines the portion of a channel occupied at a time by the users. For example, 0.4E means 40% of the channel is occupied over time. Also, offered traffic can be expressed mathematically by Little's law as follows [79]:

$$G_u = \lambda\tau \tag{4.1}$$

where G_u is the offered traffic level of an individual user, λ is the average number of service requests per unit time and τ is the average service duration. For a system with N users, the total offered traffic is the product of the individual offered traffic and the number of users:

$$G = G_u N \tag{4.2}$$

And from equations (4.1) and (4.2) the total offered traffic G is:

$$G = \lambda\tau N \tag{4.3}$$

Finally, offered traffic can also be measured in bits/sec, but in this work, Erlang is preferred to the bits/sec.

4.4.2 Throughput

Throughput is defined as the successful information delivered over a certain link in a communication system [80]. This parameter is frequently used for the performance measurement of a system offering data services. In our work, throughput is calculated upon successful reception of RACH request sent to eNB by the users associated with the system. In this research, a single RACH channel to calculate the throughput is considered, where the maximum channel capacity is 1E. The throughput is calculated as below:

$$S(E) = \frac{R_s T_{tx}}{t_{total}} \tag{4.4}$$

where $S(E)$ is throughput in Erlang, R_s is the number of successful RACH requests, in our work each successful request is defined as a successful packet reception on eNB, where all the packets have same length defined in bits, T_{tx} is the transmission time of each packet. In this work, all packets have the same transmission time because they have the same length and the transmission time is calculated from the packet length as shown in equation 4.5, and t_{total} is the period over which the throughput is calculated, which is the simulation time in this work.

$$T_{tx} = \frac{l}{C} \tag{4.5}$$

where l is the packet length in bits and C is the data rate in bits/sec. Similar to offered traffic, throughput can also be obtained in bits/sec as follows:

$$S(\frac{bits}{sec}) = \frac{R_s l}{t_{total}} \tag{4.6}$$

4.4.3 End-to-End Delay

End-to-end delay is an important time-dependent parameter, which is calculated from the time of packet generation to the time of successful packet reception at the destination node. This type of delay consists of several components, which are the queuing delay prior to transmission, scheduling delay from the MAC protocol, packet transmission duration, and radio propagation delay. Therefore, if t_s denotes the time of successful RACH transmission and t_g refers to the time of RACH request generation then the average delay D of the i-th delay is expressed as:

$$D = \frac{1}{L} \sum_{i=0}^{L} (t_s - t_g) \tag{4.7}$$

where L is the number of successful transmissions from which the delay values are calculated. The mean value of end-to-end delay is commonly used as a performance metric and so results presented in this thesis has shown the mean end-to-end delay performance as a function of the channel offered load.

4.4.4 Blocking Probability

The probability with which calls are blocked is termed the blocking probability in a cellular system. Blocking probability is also defined as Grade of Service (GOS), which is used by communication engineers to design and dimension a communication system with the maximum usage demand. This work has considered blocking probability where blocking is used to define the RACH access failure. In other words, when there will be a collision and the number of allowed retransmission has been exceeded, the request will be declared as blocked. The blocking probability P_B is described mathematically by the following equation:

$$P_B = \frac{R_B}{(R_B + R_S)} \tag{4.8}$$

where, R_B represents the total number of blocked RACH requests and $(R_B + R_S)$ represents the total number of RACH requests. Alternatively, P_B can be expressed as;

$$P_B = 1 - \frac{S(E)}{T_L} \tag{4.9}$$

where T_L is the generated traffic load.

4.4.5 Traffic Models

Traffic modelling plays a vital role in designing and analysing a communication network. In this work, traffic model is considered in order to evaluate the performance of the MAC in a cellular network that coexists with H2H and M2M devices. The concept of the traffic model was developed based on the traditional telephone network by Agner K. Erlang [81]. The model was structured based on queuing and prediction of allowed blocking probability. This can be expressed mathematically by the Erlang B formula shown below [82]:

$$P_b = \frac{\frac{G^m}{m!}}{\sum_{i=0}^{m} \frac{G^i}{i!}} \tag{4.10}$$

where P_b is the predicted allowed blocking probability (which the system can tolerate) m is the number of servers (e.g. telephone lines). G, as defined in equation 4.3, is a function of λ, which is the average call/packet arrivals in a given time using predicted traffic distribution. In this work, λ refers to the average RACH request arrivals in a given period of time.

In communication, different types of traffic distributions can be used to describe behaviour based on the arrival of information. The appropriate traffic model depends on the nature of the application of the information arrived.

A common traffic model called the Poisson distribution model is considered in this research, which is one of the most important models in queuing theory. Although the Poisson distribution was originally used for modelling fixed networks, it can also be used to model cellular networks. This model is well established and has been used widely in the call arrival process at the telephone exchange as it resembles typical human behaviour. It is characterised as a renewal process with exponential inter-arrival time and has the following characteristics [83]:

- Infinite number of sources
- Random traffic arrival pattern

In our work, as a traffic model we have considered Poisson for both H2H and M2M. This is because the Poisson model is well established as an accurate model of the call arrival process at the telephone exchange/eNB. In our work, we have considered different traffic loads for M2M aggregated with H2H traffic.

4.5 Summary

An overview of the communication simulation system is discussed in this chapter. Riverbed Modeller is used as a simulation tool and a reference model of the developed simulator is also presented in section 4.3.1. Also, the overall network diagram and example node structure diagrams, process state diagram models are also discussed in section 4.3.2. In section 4.3.3, we presented the pipeline stages of Riverbed Modeller. Section 4.4 includes the key metrics used to assess the performance of the simulated protocols i.e. throughput, blocking probability, and end-to-end delay of the network. Finally, the traffic model used in this thesis is discussed in 4.4.5.

Chapter 5

Two-step slotted-ALOHA RACH Access

5.1 Introduction

This chapter investigates two-step slotted-ALOHA RACH with M2M traffic which coexists with H2H. The chapter starts in section 5.2 with the introduction of the basic of two ALOHA schemes (slotted and pure) with their performance comparison. A two-step RA procedure is described in section 5.3. Section 5.4 we investigate the two-step SA-RACH model by evaluating performance using throughput, delay and blocking probability. In Section 5.5 a group-based two-step SA-RACH is investigated and performance evaluation is presented in section 5.6. Finally, section 5.7 summarises the chapter.

5.2 ALOHA Schemes

This section provides an overview of ALOHA (pure ALOHA and slotted-ALOHA) mainly focusing on slotted-ALOHA which is considered as a baseline in our work. The ALOHA schemes were developed at the University of Hawaii in 1979 and was the first multiple access data communication network [84]. ALOHA use a simple MAC strategy that allows users to share a single channel without any control by central entity [85]. Expressions of the ALOHA schemes can be derived using Poisson distribution with some simplifying assumptions. The equations originated as results this way, provide useful theoretical upper bound on the throughput. To analyse the efficiency of p-ALOHA we make the following assumptions regards to the user [78].

- all the packets are equal in length.
- There is a huge number of user/node

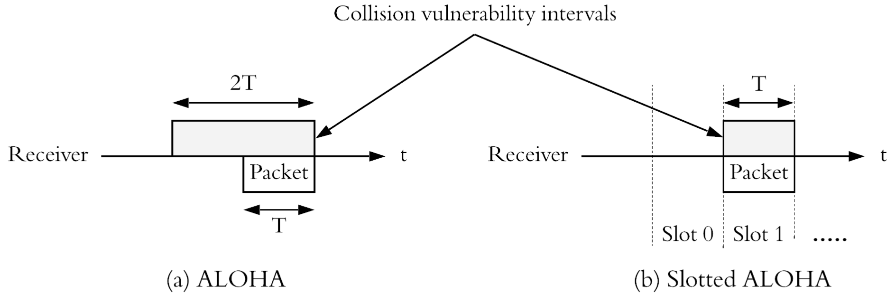

Figure 5.1: p-ALOHA showing vulnerable period of transmission, reproduced from [98]

- request arrival should follow Poisson arrival process.
- There will be lost packet due to collision fully or partially.

Pure ALOHA: Fig.5.1(a), shows how packet collision in pure ALOHA can potentially occur. If a receiver receives a packet at time t with packet duration T, and again receives another packet within the time duration $[t-T, t+T]$, collision would occur. The probability of k number of packets arriving at the receiver during time period t is [85]:

$$P_{\mathrm{k}}\{t\} = \frac{(\lambda t)^k e^{-\lambda t}}{k!} \tag{5.1}$$

where λ is the arrival rate in packet/s, and so, $1/\lambda$ is the average packet inter-arrival time where packets are generated as in the Poisson traffic model. The time duration for probable collisions is $2T$. Therefore, probability of successful packet transmission during this period t = $2T$ can be found using eq 5.1:

$$\begin{aligned} P_0\{2T\} &= \frac{(\lambda 2T)^0 e^{-\lambda 2T}}{0!} \\ &= e^{-2G} \end{aligned} \tag{5.2}$$

where G represents the amount of traffic arrival at the channel in Erlangs and on average $G = \lambda T$. Throughput Scan be calculated from the above equation by multiplicating the probability of success by G:

$$S = Ge^{-2G} \tag{5.3}$$

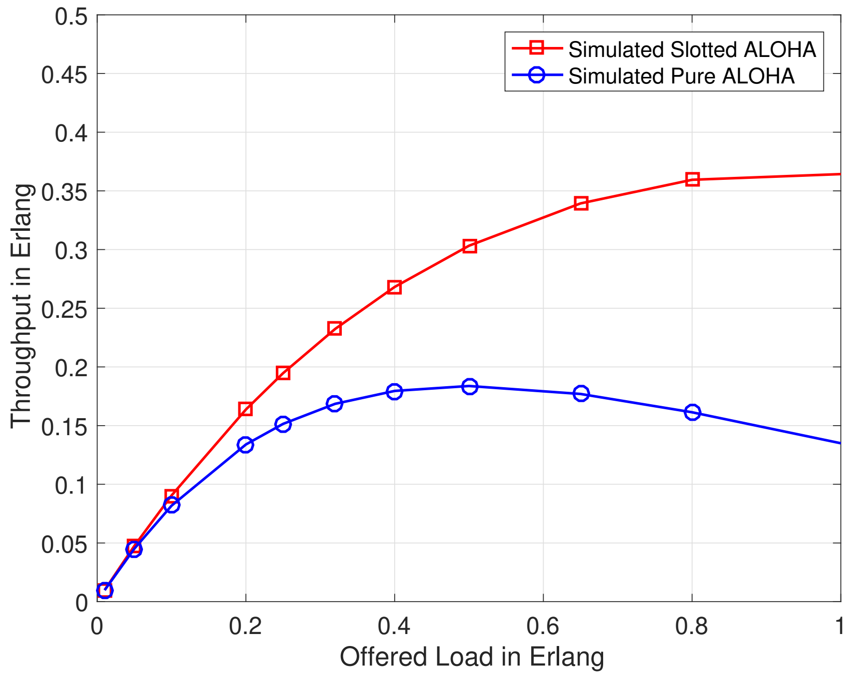

Figure 5.2: ALOHA comparison

Maximum throughput can be derived using the above equation by finding throughput where the offered traffic is 0, which results in the required offered load for maximum throughput to be:

$$G = \frac{1}{2} \tag{5.4}$$

This gives the theoretical maximum throughput of 0.1835 Erlangs when the offered traffic is 0.5 Erlangs [84].

Slotted ALOHA: Similar to pure ALOHA but with two additional assumptions, slotted ALOHA can be analysed. Here, more than one packet must be received at the same receiver within the same time slot for a collision. In this case, the collision interval is reduced from $2T$ to T as packets must overlap within the same time slot, this can be seen in Fig 5.1(b). When the probability of successful packet reception is e^{-GT} , throughput for a given offered traffic can be calculated as:

$$S = Ge^{-GT} \tag{5.5}$$

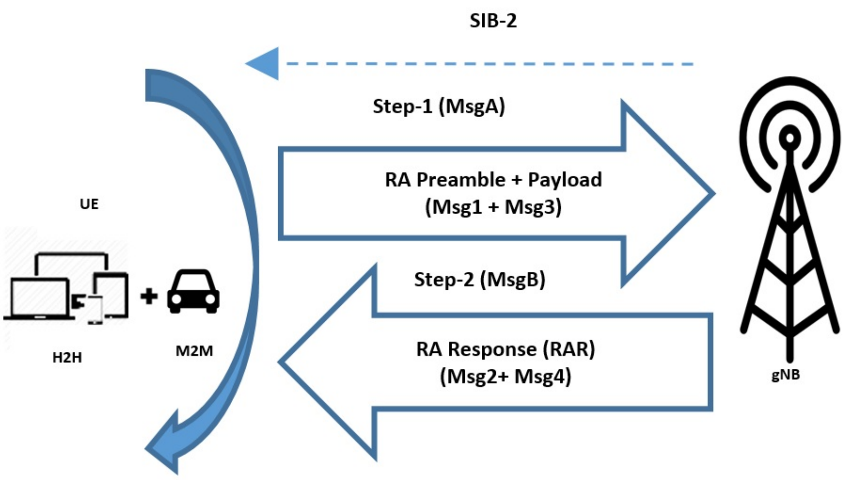

Figure 5.3: Two-step Random Access Procedure

Hence, the theoretical maximum throughput is 0.3679 Erlangs with channel capacity of 0.3679 Erlangs, when the offered traffic is 1 Erlang [84]. Figure 5.2 depicts the predicted theoretical throughput characteristics of the two ALOHA schemes.

5.3 Two-step Random Access Procedure

3GPP has recently introduced a two-step Random Access Procedure which aims to reduce signalling overhead and latency [99]. In the first step of two-step RA procedure, a UE transmits a preamble with payload, i.e., control message or data, using uplink resources randomly selected by UEs as shown in Fig.5.3. After transmitting MsgA (Msg1+Msg3) which is similar to the preamble transmission of 4-step random access (discussed in section 3.2.2), a UE waits for MsgB (Msg2+Msg4), similar to the Random Access Response (RAR) of the 4-step RA procedure. Different actions are taken by the next generation nodeB (gNB) depending on the reception status of MsgA:

- Case 1: If the gNB successfully detects the preamble and decodes the payload, it sends a successful random access response (RAR) to the UE.
- Case 2: If the gNB detects multiple identical preambles from UEs, it considers this as collision and therefore gNB transmits a backoff indication to the UEs to retry the transmission.
- Case 3: If the gNB fails to detect any preamble, there will be no RAR.

The main advantage of two-step RA procedure is reduction in latency and signalling overhead. However, with the increase in number of UEs, achieving the low latency goal becomes challenging. When a massive number of UEs compete for limited random access resources, frequent collision occurs followed by multiple random access re-attempts which results in increased delay, which is intolerable for some delay-critical UEs. In our work, we have adopted a slotted-ALOHA based two-step RA procedure where H2H and M2M traffic coexist. In the following section, we investigate the impact of M2M on two-step SA-RACH.

5.4 Impact of M2M on Two-step SA-RACH

In this work, slotted-ALOHA is used as a basis of our developed protocol because of its simplicity and the nature of the RACH channel in which the access is restricted to slots. This protocol has the ability of handling multiple spatially distributed nodes, where a single channel will be assigned to all the user without involving the central entity. We investigate two-step SA-RACH for accommodating M2M devices in the existing cellular network. In the following section, impact of M2M on two-step SA-RACH is investigated.

5.4.1 Investigation of Two-step SA-RACH with retransmissions

In slotted-ALOHA RACH, due to the nature of the protocol, collision is inescapable and it causes poor throughput performance at high traffic loads. In order to maximise the chance of the request getting through, it allows retransmission after a collision happens. Retransmission help to reduce the possibility of blocking the user by giving more chances. However, it is essential to control the retransmission to make a system stable because at higher load, higher retransmission will generate more retransmitted traffic so the aggregated traffic will be high and the system will be unstable.

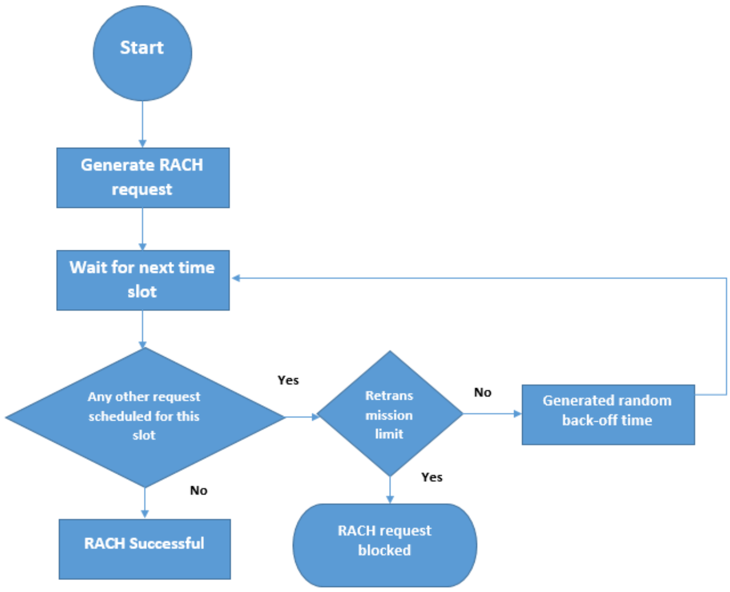

Figure 5.4: RACH request process diagram

As shown in Fig.5.4 a user generates a RACH request and needs to wait for the next time slot to send the request if that particular slot is used by another user at the same time, collision occurs. After the collision, a user checks its retransmission value and if it is minimum than max retransmission number then it will generate a random backoff window and retransmit again. If a user exceeds the maximum retransmission number, the request will drop and will need to start from the beginning. For two-step SA-RACH, we consider some assumption these assumptions represents a scenario that allows us to analyse RACH throughput of two-step SA-RACH according to the standards. These assumptions are:

- RACH request packet is equal to the slot length.
- There is a huge number of user generating request in a short interval.
- RACH request arrival should follow Poisson arrival process.
- The system is perfectly synchronised with every user and can transmit only at the beginning of a slot.
- Only one preamble available per slot which means all user sharing a single RACH.

5.4.2 Two-step SA-RACH performance evaluation

To evaluate the performance of the two-step SA-RACH, the Riverbed Modeller is considered as the simulation tool, where both H2H and M2M used the standard parameter as shown in Table 5.1. Also, the newly developed model is validated and compared using [72] and [86].

Table 5.1: Two-step SA-RACH Simulation parameters

Parameter	Values
Slot Duration	1 ms
Frame Duration	10 ms
ReTx limit	7
PRACH Config Value	8,10,12,14
No. of Preamble	1
Max Backoff Value	14

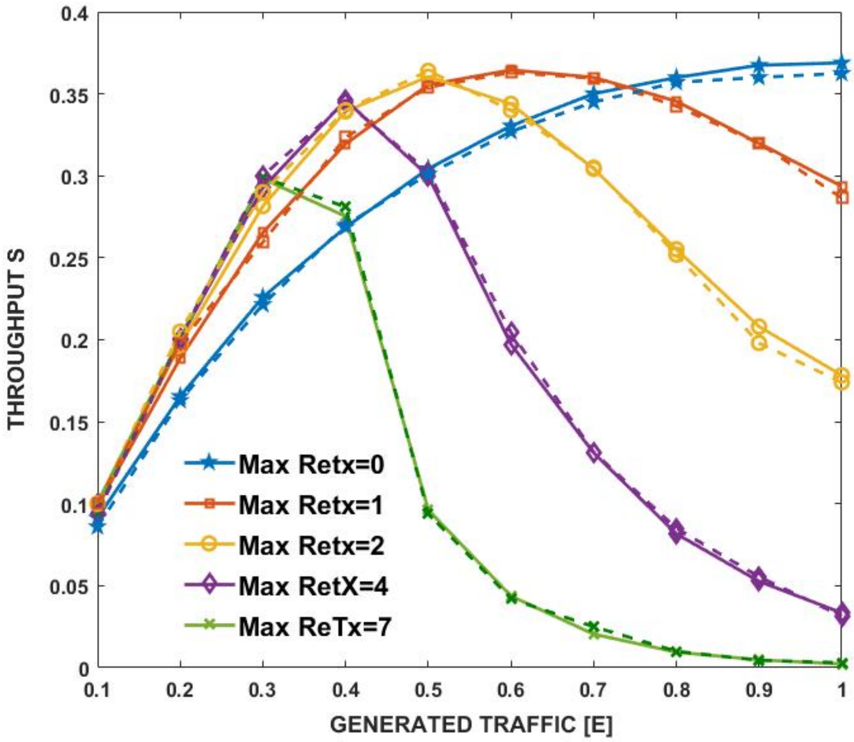

Figure 5.5: RACH throughput against generated traffic (Solid line represents our result and dashed line represent the result in [86])

Fig.5.5 shows the RACH throughput performance at different retransmission values (1, 2, 4 and 7) with a maximum backoff interval of 14 slots within which a user selects a slot at random. The results show that at low generated traffic the highest maximum number of retransmission produces better throughput. Although with the growth of the generated traffic the throughput increases up, after some point it starts to fall. For example, for a maximum retransmission value of 4, it shows an increasing trend in throughput up to a value of 0.4E. Nevertheless, at higher generated traffic it starts to fall as the retransmission traffic inject more traffic to the system, causing the traffic to exceed s-ALOHA capacity.

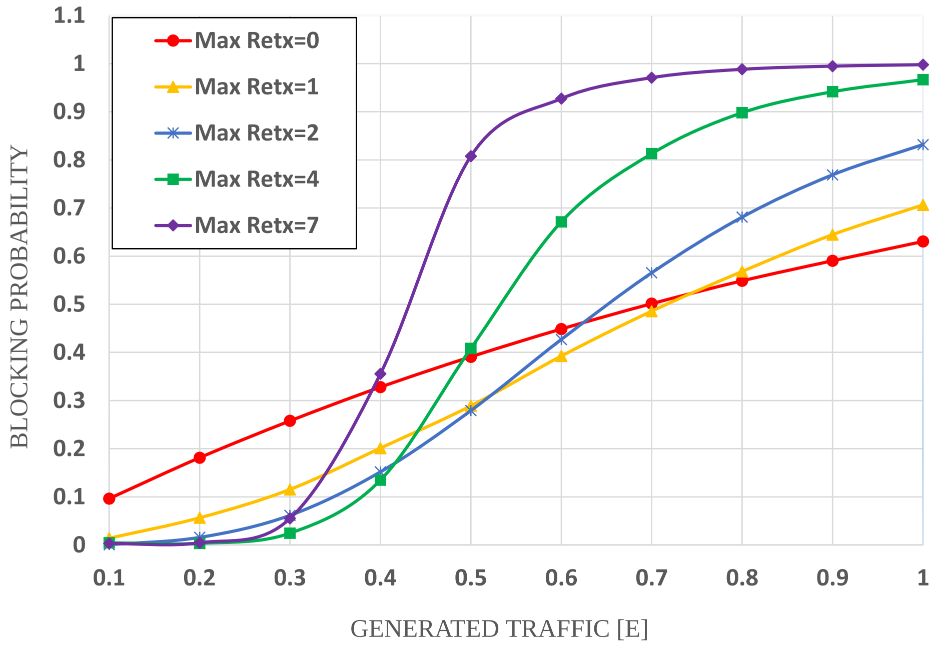

Figure 5.6: Blocking probability against generated traffic for different retransmission number with a random retransmission interval of max backoff value fixed at 14 RACH slots

Fig.5.6 shows the blocking probability, which agrees with the behaviour of the throughput curves explained above. As shown, at low generated traffic levels, the maximum number of retransmission reduces the blocking significantly. On the other hand, at higher traffic levels, blocking probability increases as the number of retransmission increases. This is because when the generated traffic is high the system pushed beyond slotted-ALOHA capacity, so more collision occur resulting into more retransmission, hence the blocking probability becomes high.

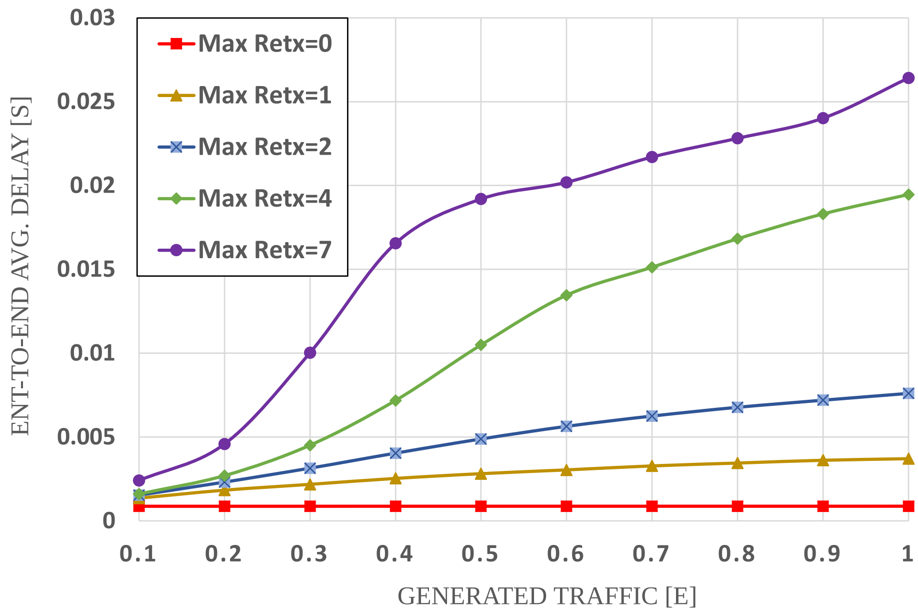

Figure 5.7: Average end-to-end Delay against generated traffic for different retransmission number with a random retransmission interval of max backoff value fixed at 14 RACH slots

Average end-to-end delay is another parameter used in our work to describe the behaviour of RACH access at a different number of transmissions. In general, Fig.5.7 showed that the average end-to-end delay is increasing in nature with respect to generated traffic growth. Fig.5.7 showed that when there is no retransmission (max retx=0), the delay is almost the same throughout the range of generated traffic, but the delay increases with the increase in the maximum number of retransmission, where at maximum retransmission value 7 the delay is also maximum. Therefore, we can see that there is a trade-off between RACH throughput and average end-to-end delay with regards to the maximum number of retransmissions.

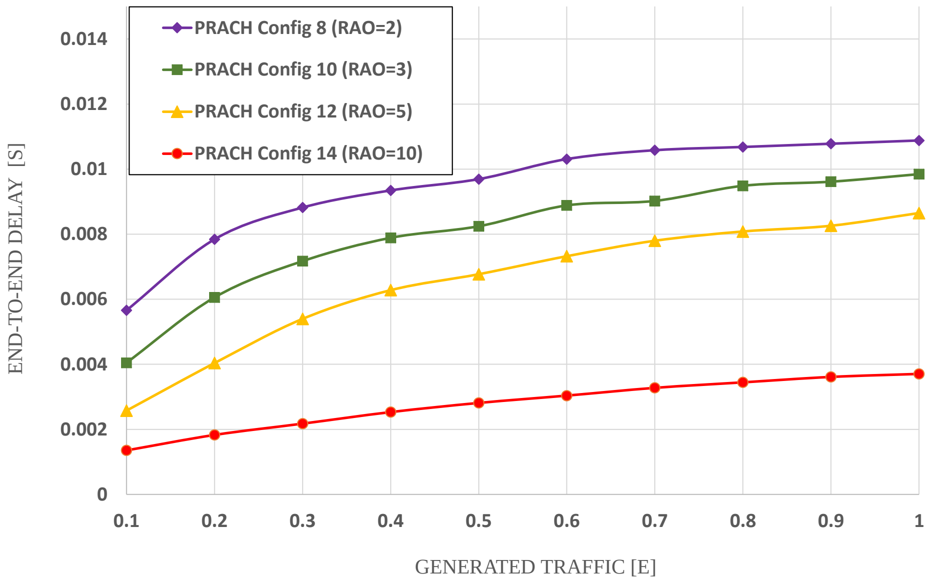

Figure 5.8: Average end-to-end against generated traffic for different PRACH configuration values with a random retransmission interval of max backoff value fixed at 14 RACH slots

The impact of RACH configuration index (CI) on the average access delay is illustrated in Fig.5.8 In particular, as shown in Fig.5.8 the average access delay increases with increasing traffic load. Besides this, it also shows that at a higher configuration index, corresponding to a higher number of RA opportunities (RAO) per frame; the access delay reduces compared to a lower number of configuration index.

The retransmission interval width is the second parameter used to determine the retransmission behaviour in this work. A high interval reduces the probability of more than one user retransmitting at the same time slot that could cause another collision. As shown in Fig.5.9 the retransmission interval has an impact on average end-to-end delay. In general, for all interval values delay increases with the increase of generated traffic level. For example, retransmission interval width value 3 (3 RACH slot) shows an increasing trend in delay and becomes stable at a higher load with a value of about 0.01s. On the other hand, retransmission interval width value 50 (50 RACH slot) shows a different characteristic, it shows that the delay increases with respect to generated traffic growth and becomes large compared to interval width 3. However, it is clear from the above Fig.5.9 that at a low generated traffic level, the delay introduced for the high retransmission interval is accepted for some delay critical applications but at a higher traffic load, the delay is intolerable.

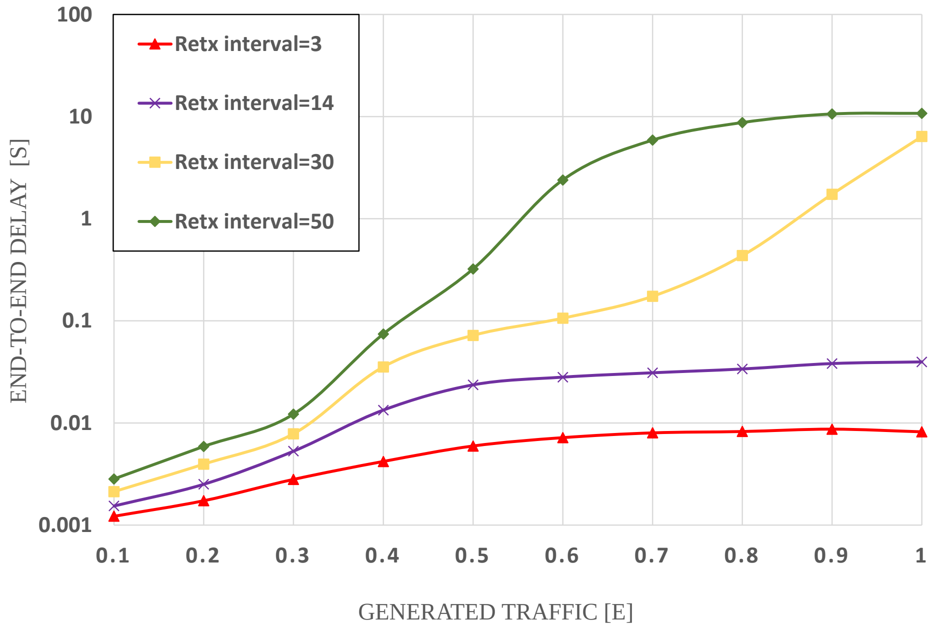

Figure 5.9: Average end-to-end delay against generated traffic for different retransmission interval values with a fixed retransmission value 7

As can be seen from Fig.5.5, the throughput performance of two-step SA-RACH is inadequate due to the limited channel capacity, which aggregates with the retransmission of the contended traffic. Consequently, the traffic surpasses the channel capacity causing the system to become unstable. Although the slotted-ALOHA for RACH access proves to be unstable as stated above, it is efficient enough in terms of H2H communication and the reason lies in the dimensioning of the system and regularity of the H2H traffic as the RACH request falls within the slotted-ALOHA throughput capacity. On the contrary, using a shared two-step SA-RACH for a massive number of M2M user with different QoS requirements will not be efficient. Because a massive number of M2M traffic has the potential capability to cause the RACH overload, it will affect the slotted-ALOHA for practical use. To conclude, it is necessary to develop a congestion control mechanism and a separate group based RACH for M2M user with specific QoS requirements.

5.5 Group-based two-step SA-RACH

When a massive number of H2H and M2M devices try to access the network simultaneously, due to the limited RACH resources, there will be RACH resource shortage. This will lead to collisions and as a result, the RACH becomes unstable and causes an excessive delay and the system becomes overloaded. As investigated in section 5.4.1, it is clear that the two-step SA-RACH performs differently in different scenarios with different sets of parameters. In low load scenarios, two-step RACH performs well in both H2H and M2M UEs. On the contrary, in high load scenarios, two-step RACH performance degrades which is unacceptable for some application scenarios and specific user groups. In order to fulfil the requirements of different groups (DC and NDC), it is crucial to choose suitable parameters so that two-step RACH can provide better performance. In our work, we have combined two random access approaches (Group-Based RA, Separate Resource Allocation based RA) to address these issues with two-step RACH. In this two-step group-based SA-RACH model, H2H and M2M have their own separate groups. The M2M user is further divided into two groups:

- Delay Critical (DC) and
- Non-Delay critical (NDC)

The motivation behind the approach is to mitigate the delay issue for M2M delay-critical UEs. We have considered a single preamble based shared RACH which means that there is only one RACH opportunity per RA slot. Instead of using regular preamble, we have adopted a coded-preamble by which eNB can identify the user group. Additionally, a load aware dynamic RACH configuration model is adopted instead of a fixed value RACH model. We have also considered two different types of scenarios: low load and high load. In low load scenarios, both DC and NDC (Dual user mode) have the same parameters. On the contrary, in the high load scenarios, the parameter set for the DC user group is changed in a dynamic manner with respect to the network load condition to minimise the collision and reduce the delay for DC-UEs. This scheme is able to adjust the PRACH configure index, backoff Interval (BI) value, and max Retx value according to the load to satisfy the delay requirements for DC-UEs.

5.6 Performance Evaluation of Group-based two-step SA-RACH

5.6.1 Simulation Scenario and parameters

To evaluate the performance of proposed Group-based two-step RACH schemes some assumptions for both groups of UEs are considered to represent the scenario.

These assumptions are:

- All packet lengths are equal to the length of the slot.
- RACH request arrivals follow the Poisson arrival process.
- The system is perfectly synchronised with every user transmitting at the beginning of a slot.
- All users share a single RACH (one preamble available).

Based on the assumptions, the simulation is modelled with the introduction of the RACH request generation from each user group (H2H and M2M) following exponentially distributed inter-arrival times with a mean inter-arrival time τ_{ia} determined by the traffic load G as shown below.

$$\tau_{ia} = P_{len}N/GT_{rate} \tag{5.6}$$

Where P_{len} is the preamble length/packet length in bits, N is the total number of users in the system, G is the traffic load in Erlangs and T_{rate} is the transmission rate in bits/sec.

The collaboration between the user groups (H2H & M2M) is controlled by providing individual loads to each group. Therefore, the total generated traffic in the system using the load sharing is expressed as:

$$G_{Total} = G_{H2H} + G_{M2M} \tag{5.7}$$

Where G_{Total} is the total generated traffic, G_{H2H} and G_{M2M} represent traffic generated by H2H and M2M respectively.

To evaluate the performance of the Group-based two-step SA-RACH, we have used the Riverbed Modeller as the simulation tool. The simulation is performed in the LTE scenario of a single cell, with an eNB. The basic parameters set up for the random access procedure are as shown in Table 5.2.

Table 5.2: Simulation parameters for Dual user group

Parameter	DC-UEs	NDC-UEs
RAO	5 Per frame	5 Per Frame
Slot Duration	1 ms	1ms
Frame Duration	10 ms	10 ms
ReTx limit	7	7,3
PRACH Config Value	12	12,13
No. of Preamble	1	1
Max Backoff Value	5	20,5

5.6.2 Simulation results and discussion

As can be seen from Fig.5.7, the max Retx value impacts the end-to-end delay. In low load conditions, the minimum value of max Retx (in our work minimum value of max Retx=0) achieves low delay. On the other hand, in high load conditions, all UEs with max Retx value (in our work max value of max Retx=7) will cause very high delay as after each collision, all UEs will retry maximum times. Therefore, adjusting the max Retx value is crucial in high load scenarios.

For DC-UEs, the value for BI and max Retx value stays the same in both low load and high load scenarios. On the contrary, for NDC-UEs that are delay-tolerant, in low load scenarios, the value for BI and max Retx value is the same as DC-UEs. However, in high load scenarios, the BI value is increased and the max Retx value is decreased to achieve the delay requirements. Another important parameter is PRACH config index which determines the number of slots available per LTE frame. From Fig.5.8, it is seen that a higher PRACH config index reduces the access delay compared to a lower PRACH config index. Taking this into consideration, we have used a similar PRACH config index in low load scenarios, whereas, in high load scenarios, we have used a different PRACH config index for DC-UEs and NDC-UEs.

Fig.5.10 and 5.11 shows the average end to end delay versus the offered load for different user groups. Here, the single user group (M2M) is considered as DC-UEs and the single user group (H2H) is considered as NDC-UEs. When the network load increases, the end-to-end delay increases for NDC-UEs. When the load is less than 0.4E, it is considered as low load scenario, and the same set of parameters are assigned to both user groups. However, in high load scenarios (load>0.4E), the RACH parameters are changed dynamically for NDC-UEs. In high load, DC-UEs use PRACH config index 13 (1, 3, 5, 7, and 9), whereas NDC-UEs use PRACH config index 12 (0, 2, 4, 6 and 8). The result shows that the delay for M2M user

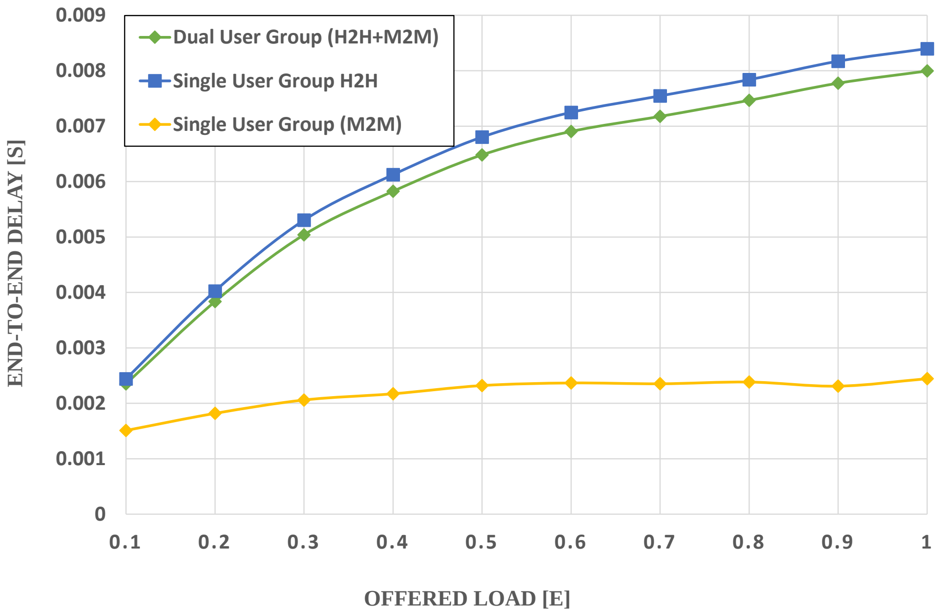

Figure 5.10: Average end-to-end delay against offered load (Fixed M2M load and varying H2H load)

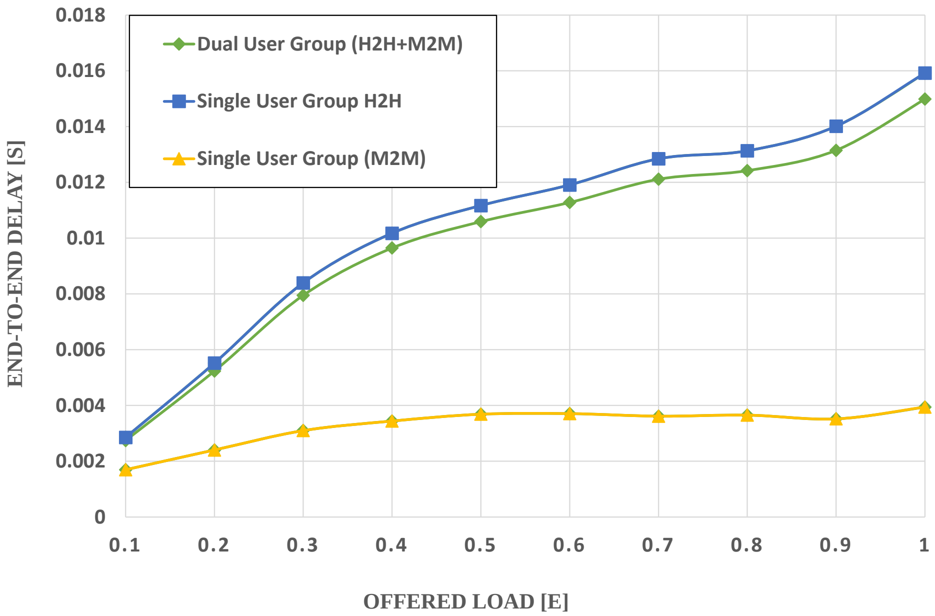

Figure 5.11: Average end-to-end delay against offered load (Fixed H2H load and varying M2M load)

group becomes stable because they avoid collision with H2H and have more access in slots with different sets of BI and Retx value.

In the above study, it is investigated that a dynamic RACH configuration approach for LTE reduce the massive access problem and also can reduce the access delay of a specific user group called M2M DC-UE with high priority without affecting the total number of accesses.

5.7 Summary

This chapter has presented an overview of the existing RACH access scheme. Comparison between pure ALOHA and slotted ALOHA schemes are presented. Two-step SA-RACH instability due to M2M is investigated and it was realised that SA-RACH scheme has shown difficulties in supporting additional M2M traffic on the existing cellular network. To address this important issue a group-based two-step SA-RACH is proposed for specific user group and is shows effectiveness by keeping the end-to-end delay minimum in both higher load scenarios. Although a group based RACH is proposed in this work, this method is not dynamic and only works with the eNB broadcast message. In order to mitigate this problem in our work in the next chapter, we have introduced a dynamic prioritised RACH.

Chapter 6

Prioritised Dynamic RACH (PD-RACH) Scheme for DC-UEs

6.1 Introduction

The main issue of RACH overload, which is massive access requests on RACH with a massive number of M2M devices was analysed in the previous chapter. The issue indicates that the existing RACH access scheme will not be efficient for the additional M2M traffic over the cellular network. It is also seen that at a peak traffic load where the number of access requests is maximum, the standard LTE random access mechanism suffers congestion due to the high probability of collision and this causes excessive time delay. To mitigate this issue in chapter 5, a group-based two-step SA-RACH is proposed for a specific user group and it shows effectiveness by keeping the end-to-end delay minimum in both higher and lower load scenarios. The limitation of this method is that the proposed scheme is not dynamic and only works with the eNB broadcast message. Therefore a dynamic scheme is proposed in this chapter in order to manage the congestion issue. In this work, a priority-based RACH scheme is proposed which aims at solving delay issue for a specific user group by proposing a Prioritised Dynamic Random Access scheme (PD-RACH). In section 6.2 the implementation details of the PD-RACH scheme are presented with the PD-RACH algorithm. In section 6.3, the performance evaluation of PD-RACH schemes is investigated. Finally, section 6.4 summarises the chapter.

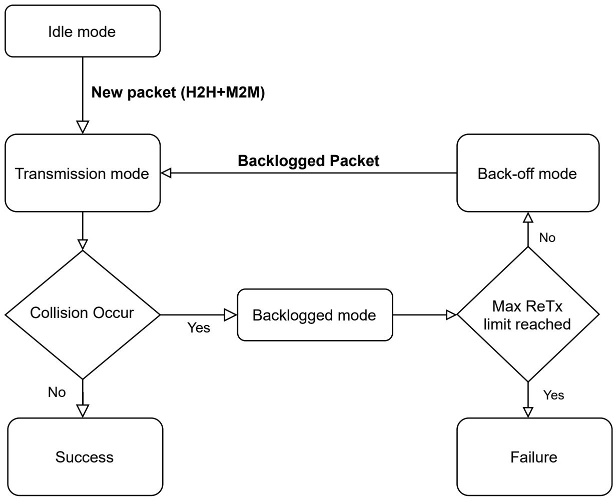

Figure 6.1: RACH request process of a cellular system.

6.2 Prioritised Dynamic RACH (PD-RACH) Scheme

A shared RACH with the coexistence of H2H and M2M UEs is considered in this work. Also, the M2M users are categorised into two groups: delay-critical (DC-UEs) and non-delay-critical (NDC-UEs). We propose a Priority-based slotted-ALOHA Dynamic RACH (PD-RACH) scheme to control the interaction and the delay for the specific delay-critical user group, whereas in the traditional slotted-ALOHA RACH, an equal opportunity based fixed RACH is used for both H2H and M2M users. The proposed scheme provides a two-step prioritisation mechanism and includes a combination of dynamic allocation of RACH resources and a group based specific backoff by utilising random access cycle-based information. The eNB can monitor this information to adjust RA resources and other parameters like backoff and Max-Retransmission value in order to minimise delay and provide priority to delay-critical UEs over non-delay- critical UEs. Further, we analyse whether our proposed scheme and simulations results validate the effectiveness of the proposed scheme compared to the conventional scheme.

6.2.1 SA-RACH scheme and Impact of M2M device

A SA-RACH request process of combined H2H and M2M (DC/NDC) devices in a cellular system is shown in Fig. 6.1. A user-generates a RACH request and needs to wait for the next time slot or transmission mode to send the request. If that particular slot is used by another user at the same time, a collision occurs and this collision phenomenon is inescapable due to the nature of this protocol. After a collision, a user checks its retransmission value and if it is within the max retransmission (MaxReTx) limit then it will generate a random backoff value and retransmit again, where this retransmitted request is called a backlogged request. The cellular system allows retransmission in order to maximise the chance of the request getting through which helps to reduce the possibility of blocking by giving more chances to the user. If a user exceeds the maximum retransmission number, the request will be dropped and needs to start from the beginning. However, with a large number of M2M devices, the activity of the channel may be very high because M2M devices are capable of originating a huge number of access requests in a short time interval. This will cause a large number of collisions which will produce more backlogged requests, and the aggregated traffic will be very high which drives the system beyond its capacity. As a result, the system will be unstable. This is why in such an overloaded scenario a prioritised RACH is essential to control the system for delay-critical UEs in order to satisfy the delay requirements by providing a suitable backoff scheme as well as controlling the retransmission number.

6.2.2 Proposed PD-RACH scheme for delay critical M2M UEs

The eNB has the ability to control RACH access by restricting or allocating RACH resources. To do so an eNB needs to have knowledge of the network condition. The available information to an eNB is limited to the number of available preambles, the total number of successful or collided preambles and number of devices registered. Even though the eNB has the information about the total number of UEs in a network it is very difficult to handle the RA process because of M2M devices submitting a huge number of access requests in a short period of time and it is much more difficult when H2H and M2M are combined in a shared RACH process. In this work, a single preamble RACH is considered with a preamble coding technique, preamble code=0 for non-priority non-delay critical user group (NDC-UEs) and preamble code=1 for the delay-critical priority user group (DC-UEs). The main idea behind this preamble coding is to identify different user groups in order to provide priority. We employ a frame-based statistic strategy where a frame based Random Access cycle, RA_{cycle} (RA_{cycle} < SIB2 Broadcast time) is used (described in Algorithm 1) and after each RA_{cycle}, the preamble collision probability for overall UEs and for delay-critical UEs are calculated using (6.1) and (6.2).

$$P_C = \frac{total\, preamble\, collision\, in\, RA_{cycle}\,(overall)}{Total\, RAO\, in\, RA_{cycle}}, \tag{6.1}$$

where, P_C= Preamble collision probability (overall)

$$\text{Total preamble collision in}\, RA_{cycle} = \text{No. of total attempts} \\ -\text{No.of successful RACH attempts}$$

$$\text{Total RAO in}\, RA_{cycle} = \frac{RA_{cycle}}{\text{Frame duration}} \cdot \text{RAO per frame}$$

Also, the preamble collision probability for delay-critical UEs (DC-UEs) in Random Access cycle time RA_{cycle} can be calculated using the coded preamble technique.

$$P_{C_{DC}} = \frac{total\, preamble\, collision\, in\, RA_{cycle}(DC-UE)}{Total\, RAO\, in\, RA_{cycle}} \tag{6.2}$$

where, $P_{C_{DC}}$= Preamble collision probability (DC-UE)

These preamble collision values (P_C and $P_{C_{DC}}$) are compared with predefined preamble collision thresholds ($Pc_{threshold(min)}$, $Pc_{threshold(max)}$ and $P_{DC_{threshold}}$) in order to provide a two-stage prioritisation where first stage is defined as priority with a non-emergency stage and the other is defined as priority with an emergency stage. These priority and emergency stages are enabled using Priority and Emergency flags. If the preamble collision values (P_C) is less than $Pc_{threshold(min)}$, both Priority and Emergency flags are stated as FALSE and in this case both the delay critical and non-delay critical user group equal right with same RACH configuration parameters (*PrachConfigIndex*, Max retransmission value, backoff period) for the random access process. On the other hand, when the preamble collision probability P_C is greater than $Pc_{threshold(min)}$ and less than $Pc_{threshold(max)}$, the priority flag will be set to TRUE which indicates the first stage priority mode for DC-UEs. In this stage, a higher *PrachConfigIndex* is assigned to the delay-critical group in order to provide more RAO per frame, and frame-based backoff will be assigned instead of slot based backoff for NDC-UEs. When the Priority Flag is set to TRUE, one further step will be checked, if $P_{C_{DC}} < P_{DC_{threshold}}$ then the Emergency flag will set to FALSE, otherwise the Emergency flag will set to TRUE and this state is called the Emergency state. In the emergency state, in order to minimise the collision probability, we restrict NDC-UEs by limiting the maximum retransmission number (no backlogged request for NDC-UEs will be accepted, only newly generated requests will be accepted). In an emergency state, when the RACH is declared as congested and the preamble collision probability is already exceeded the $Pc_{threshold(max)}$ in order to provide priority to the DC-UEs, backlogged request for NDC-UEs is not accepted. Accepting backlogged request will increase the collision probability and for every collision

Algorithm 1 Preamble collision estimation based Dynamic RACH Configuration for DC-UEs

Initialisation:
T $\leftarrow$ 0, All Flag=FALSE, RA_{cycle}
Set initial RACH Paremeter:
Backoff Indicator, MaxReTX, $PrachConfigIndex$
eNB Broadcast initial RACH Paremeter via SIB2
for $T \leftarrow 0$ to T **do**
  if $T == RA_{\text{cycle}}$ **then**
    Calculate P_C and $P_{C_{DC}}$ via eqn. (6.1 and 6.2)
    UpdateRA_{cycle}
    if $P_C < Pc_{threshold(min)}$ **then**
      Priority Flag = FALSE
      Emergency Flag = FALSE
    end if
  else if $Pc_{threshold(max)} > P_C > Pc_{thrshold(min)}$ **then**
    Priority Flag = TRUE
    if $P_{C_{DC}} < P_{DC_{threshold}}$ **then**
      Emergency Flag = FALSE
    else
      Emergency Flag = TRUE
    end if
  else if $P_C > Pc_{threshold(max)}$ **then**
    Priority Flag = TRUE
    Emergency Flag = TRUE
  end if
  if $T ==$ eNB Broadcast time **then**
    Set Flag & Broadcast SIB2
  end if
end for

there will be a retransmission request which will be added to the RACH channel. Accepting only the newly generated request will minimise the interaction between DC-UEs and NDC-UEs, which eventually will minimise the collision probability for both user groups. Another case arises when the overall preamble collision probability P_C is greater than $Pc_{threshold(max)}$ both the Priority and Emergency flags will set to TRUE and this stage is called the Priority-with Emergency stage. Finally, eNB will set a group-based specific RACH configuration parameters ($PrachConfigIndex$, Max retransmission value, backoff period) based on flag indicator and will broadcast via System information block 2 (SIB2).

6.3 Performance Evaluation of Proposed PD-RACH

6.3.1 Simulation Scenario and parameters

To evaluate the performance of SA-RACH and proposed PD-RACH schemes some assumptions for both groups of UEs are considered to represent the scenario.

These assumptions are:

- All packet lengths are equal to the length of the slot.
- RACH request arrivals follow the Poisson arrival process.
- The system is perfectly synchronised with every user transmitting at the beginning of a slot.
- A RACH request will be unsuccessful only due to the collision, no noise or interference is considered in our work.
- All users share a single RACH (one preamble available).

Based on the assumptions, the simulation is modelled with the introduction of the RACH request generation from each user group (H2H and M2M) following exponentially distributed inter-arrival times with a mean inter-arrival time τ_{ia} determined by the traffic load G as shown below.

$$\tau_{ia} = P_{len}N/GT_{rate} \tag{6.3}$$

Where P_{len} is the preamble length/packet length in bits, N is the total number of users in the system, G is the traffic load in Erlangs and T_{rate} is the transmission rate in bits/sec.

The collaboration between the user groups (H2H & M2M) is controlled by providing individual loads to each group. We used a fixed load for H2H traffic at 0.1E and a varying load for M2M traffic from 0.02E to 0.2E to achieve slotted-ALOHA channel capacity. Therefore, the total generated traffic in the system using the load sharing is expressed as:

$$G_{Total} = G_{H2H} + G_{M2M} \tag{6.4}$$

Where G_{Total} is the total generated traffic, G_{H2H} and G_{M2M} represent traffic generated by H2H and M2M respectively.

6.3.2 Simulation results and discussion

In order to show the effectiveness of our new scheme, the performance of the conventional fixed RACH and proposed dynamic RACH scheme are compared for different scenarios based on the simulation parameters presented in Table 6.1.

Table 6.1: Simulation parameters for PD-RACH

Parameter	**Value**
RA Slot period	1ms
Frame Duration	10ms
Retransmission limit	5
PRACH configuration index	10,12,14
Number of Preamble sequence	1
Preamble format	1
RAO per frame	3,5,10
RAR Window size	5ms
Back-off period	20 slots, 10/20 frames
Total No. of UEs	100
No. of Delay-critical UEs	25
DC-UEs ratio	25%
Preamble collision threshold (overall) $Pc_{threshold(min)}$ and $Pc_{threshold(max)}$	0.3 and 0.5
Preamble collision threshold for DC-UEs $P_{DC_{threshold}}$	0.3
Random Access cycle (RA_{cycle})	40ms

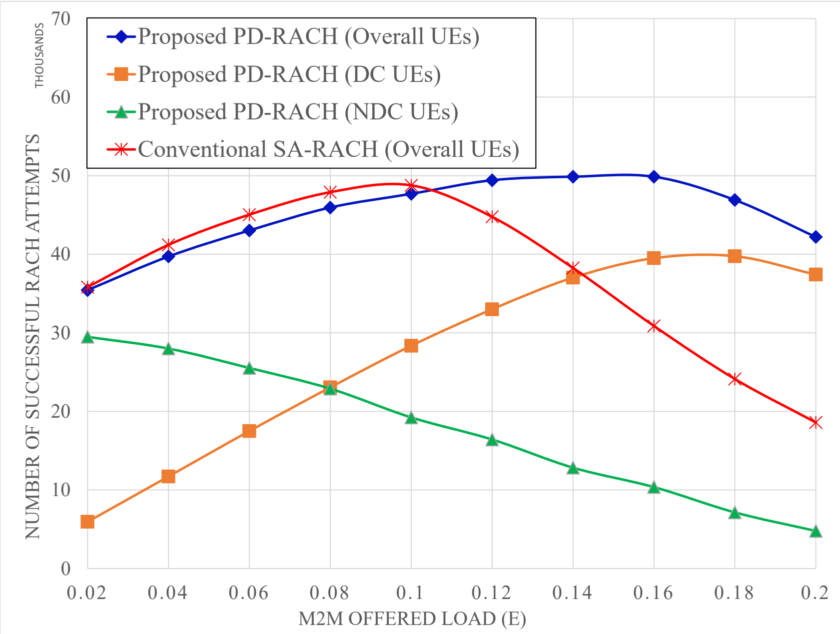

Figure 6.2: No. of successful RACH attempts vs. M2M Offered load.

Fig. 6.2 represents the successful RACH attempts versus M2M offered load. It shows in Fig.6.2 that when the offered load for both H2H and M2M is equal to or less than 0.1E, conventional RACH access performs slightly better than the proposed scheme because in conventional fixed RACH, both user groups are using a higher value of $PrachConfigIndex$ whereas in the proposed dynamic RACH scheme, both user groups will start with lower value of $PrachConfigIndex$ and will increase based on load and priority condition. However when the M2M offered load exceeds 0.1E with the conventional scheme, the number of successful RACH access attempts start decreasing immediately because with conventional RACH at high offered loads such as 0.2E, the probability of collision becomes high as a result the number of retransmission also increasing and the aggregated traffic becoming very high, exceeding the channel capacity of slotted-ALOHA. On the other hand, the proposed scheme achieves a higher number of successful RACH access attempts even at high values of offered load by minimising the collision using dynamic backoff and configuration index and it is clear from Fig. 6.2 that the prioritised dynamic RACH scheme is very effective for DC-UEs even at higher loads.

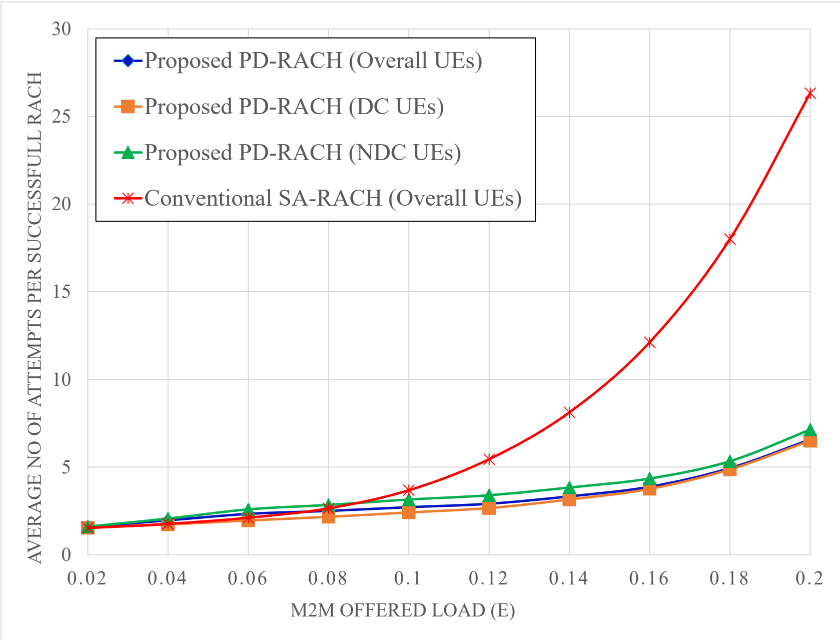

Figure 6.3: Avg. no. of attempts per successful RACH vs. M2M offered load.

Fig. 6.3 shows the average number of attempts per successful RACH access versus M2M offered load. It shows that when the offered load for both H2H and M2M is equal to or less than 0.1E, the average number of attempts per successful RACH access is at a minimum, whereas this number increases dramatically with increasing load value. When M2M load is 0.2E the number of collision increase and both user group tries their fixed maximum retransmission number using a slot based backoff which leads the maximum number of tries to reach 25 times which is intolerable for our delay-critical priority user groups. However, the performance of our proposed scheme shows that for both DC and NDC-UEs the number of average tries per successful RACH access attempts is less than the conventional scheme not only in low load condition but also in high load scenario, it keeps this value a low rate.

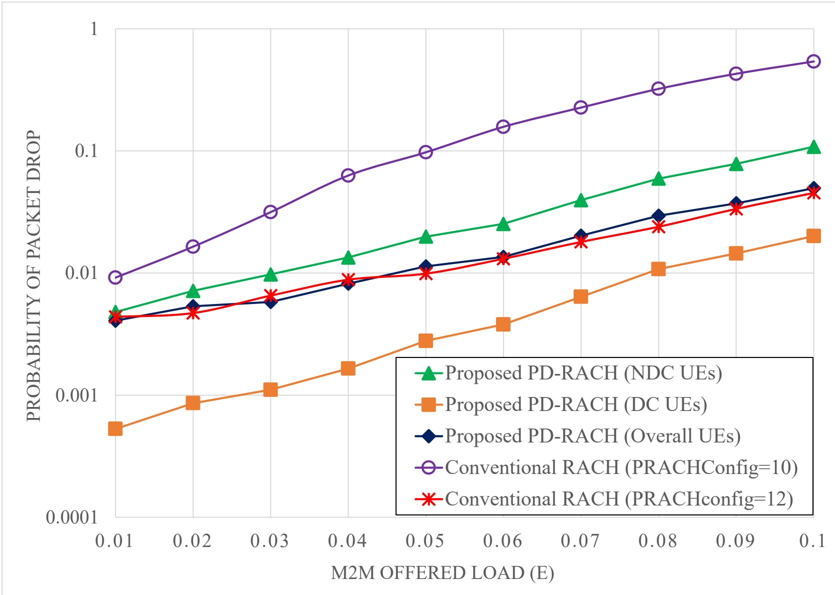

Figure 6.4: Probability of packet drop vs. M2M offered load.

Fig.6.4 shows the packet dropping probability versus M2M offered load. In this work, to evaluate the performance well within the slotted-ALOHA channel capacity, we vary the M2M offered load up to 0.1E. We also consider two different $PrachConfigIndex$ values for the conventional RACH access scheme to compare to our proposed scheme. A higher bound of $PrachConfigIndex$=12 is used where the number of Random Access Opportunities (RAO) is 5 per frame, and a lower bound of $PrachConfigIndex$=10 where RAO=3 per frame. As we can see from Fig.6.4, the conventional RACH with $PrachConfigIndex$=10 performs well under lightly loaded conditions, but under heavier loads, the performance is poorer. This is because the increased M2M load and corresponding higher overall load results in a significant number of collisions due to the limited number of random access opportunities available in the frame. On the other hand, conventional RACH with $PrachConfigIndex$=12 performs well under both high and low load conditions, but it wastes unnecessary resources for random access attempts when the channel is lightly loaded. The use of a dynamic $PrachConfigIndex$ in our proposed scheme combined with dynamic RACH parameters (maximum retransmission number and back-off values) can overcome these limitations of the fixed schemes and provide good performance for both delay-critical and non-delay-critical users. From Fig. 6.4, we can see that for delay-critical users the priority scheme outperforms both conventional RACH schemes in high and low load conditions, and the dropping probability for all users using the proposed scheme is similar to that achieved with $PrachConfigIndex$=12. In our work, even though the non-delay critical users use

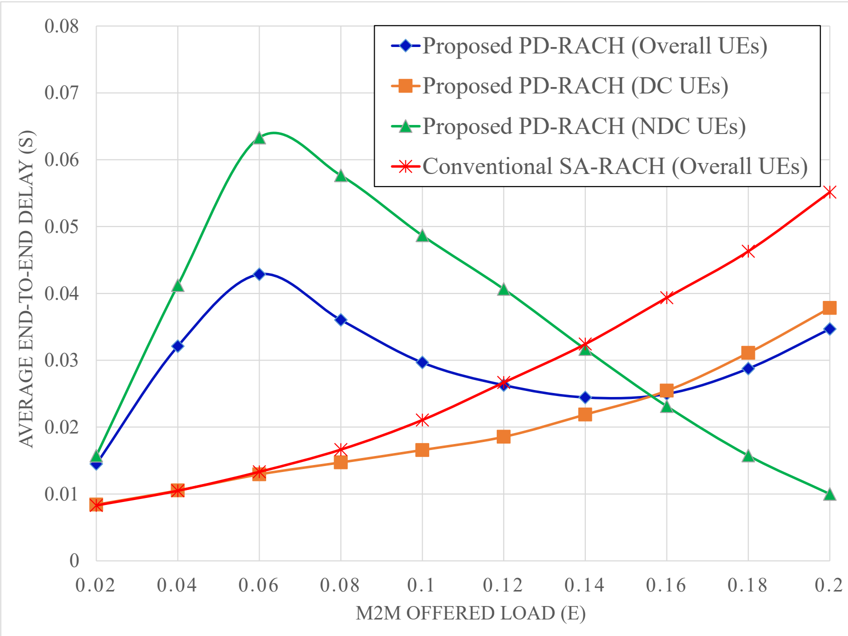

Figure 6.5: Average end-to-end delay vs. M2M offered load.

PrachConfigIndex=10, the dropping probability is much lower than the conventional RACH with *PrachConfigIndex*=10 because of the use of the dynamic RACH parameters to restrict the RACH access in higher load conditions, which reduces the number of collisions.

Fig. 6.5 shows end-to-end delay performance plotted against the M2M offered loads. As we can see for the conventional SA-RACH process, while the load for M2M is increasing, the overall delay is also increasing and at maximum load the delay is about 0.06s. In order to provide priority, a slot-based backoff is considered for DC-UEs, if a DC-UE experiences collision it can perform retransmission in the next available slot within the same frame. On the other hand, for NDC-UEs a frame-based backoff is adopted where an NDC-UEs can only perform retransmission in the next frame. With the proposed dynamic scheme, NDC-UEs initially experience a high delay because when the offered load for M2M increases the interaction between M2M and H2H also increases which leads to collision and NDC-UEs use frame-based backoff for retransmission which results in high delay. But at high loads, the delay decreases significantly because the proposed mechanism handles the RACH request and minimises the collision by limiting the interaction between user groups by providing dynamic resource allocation. For delay-critical UEs, our proposed algorithm outperforms and shows that even as the M2M load is increased the prioritised dynamic RACH scheme is able to keep the delay lower than the conventional scheme.

We have presented a performance evaluation of contention-based RACH access, where delay-critical and non-delay-critical devices coexist. It has been shown that in the case of a lightly loaded traffic scenario, use of a fixed parameter based RA scheme could satisfy the delay requirements but when the traffic is heavily loaded, the standard RACH access mechanism suffers congestion due to the high probability of collision and causes excessive delay which is intolerable for delay-critical UEs. To solve this problem, we have proposed a Prioritised Dynamic RACH (PD-RACH) scheme lies in the full consideration of the priority user group called delay-critical user group (DC-UEs). The simulation results show that in highly loaded traffic scenarios by dynamically adjusting the RACH parameter using the proposed preamble collision estimation based dynamic RACH scheme could greatly decrease access delay and a significant gain in the number of successful attempts for delay-critical UEs as compared to the conventional fixed RACH.

6.4 Summary

This chapter introduced a Priority Based Random Access which enables M2M users (specially Delay critical UEs) to coexist with H2H users even in a massively loaded scenario. A coded preamble based collision probability model is used to define the priority state where DC-UEs are given the highest priority. This work considered a two-stage priority model, one called priority state and another called emergency-state. The simulation results show that even in a highly loaded traffic scenario, by using this scheme a DC-UE can dynamically change its RACH parameter values given via SIB2 in order to get access in an emergency or overloaded situation. A significant decrease in access delay and a significant gain in the number of successful attempts is observed through the simulation results. Although a dynamic RACH is proposed here, which can handle a massive request in RACH in a dynamic manner, the collision between the same user group can not be avoided. Due to this, in the next chapter, we have introduced a novel Q-learning based RACH scheme for two-step prioritisation and the idea of this work is to use the Q-learning approach in an emergency scenario in order to solve the delay issue.

Chapter 7

Q-learning assisted Coded Preamble based Prioritised Dynamic RACH

7.1 Introduction

Chapter 6 analysed the impact of massive M2M over dynamic prioritised RACH and the simulation results show that even in a highly loaded traffic scenario, by using this scheme a DC-UE can dynamically change its RACH parameter values given via SIB2 in order to get access in an emergency or overloaded situation. A significant decrease in access delay and a significant gain in the number of successful attempts is observed through the simulation results. The limitation of this PD-RACH is that the collision between the same user group can not be avoided. In this chapter, to mitigate this issue as a further improvement the potential of the learning approach is investigated. Q-learning technique is used here as a means of supporting DC-UEs over the cellular network, where DC-UEs and NDC-UEs coexist. A novel coded preamble based dynamic Q-learning based RACH is proposed for further prioritisation to serve priority/required QoS to the priority user groups. The idea of this work is to use the Q-learning approach in an emergency scenario in order to solve the delay issue. In section 7.2 a brief overview of Q-learning and Q-learning assisted coded preamble based prioritised dynamic RACH (QPD-RACH) is given then in section 7.3 performance evaluation of QPD-RACH schemes is presented where single-user and the dual-user group is considered. Finally, the chapter is summarised in section 7.4.

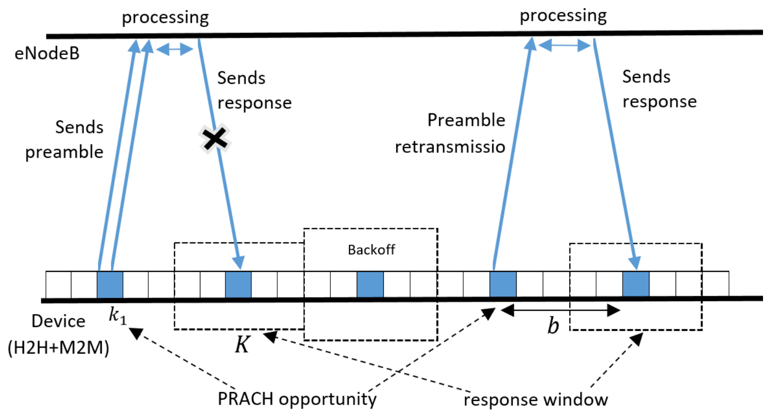

Figure 7.1: SA-Random Access Process

7.2 Q-learning assisted Prioritised Dynamic RACH

To monitor and adjust RA resource parameters as well as to set the priority, eNB uses a coded preamble based collision probability model which provides a two-step prioritisation scheme. In the first step, UE (both DC and NDC) adjust the RACH parameters like backoff value, $PrachConfigIndex$, Max-ReTx value and in the second step, a Q learning approach is applied to avoid collision among delay critical UEs in order to minimise delay and provide priority access to delay-critical UEs over non-delay-critical UEs. This work supports combined SA-RACH and Q-RACH briefly explained below.

7.2.1 Combined SA-RACH and Q-RACH

In Fig.7.1, a slotted-ALOHA based RACH Access scheme is illustrated, where users use a shared frame with one or multiple RAO, which is a number of the available slots in a frame. This RAO value depends on $PrachConfigIndex$ which is broadcast via SIB2 periodically. In the RACH process, upon reception of the SIB2 parameter, a user sends a request MSG1 as preamble transmission. In Fig 7.1, k represents a PRACH opportunity, in which a UE can send the preamble. If more than one UE select the same slot K and send preamble at the same time, a collision occurs. If eNB does not send a response within the response window (in figure the cross sign represents no response from eNB), and in such a case the UE needs to adopt a backoff value (which could be random or fixed) and after the backoff window, it will transmit again. This simple slotted RACH works fine in a low load/traffic scenario, but when the number of users is very large, and a massive number of access requests happen in a very short window, this mechanism becomes inefficient to handle the massive access request. As a result, congestion occurs, and even if a user tries multiple times, it will be

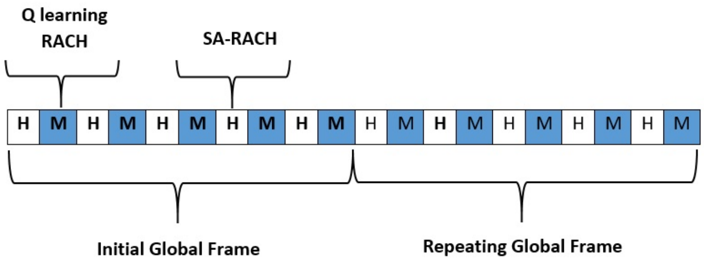

Figure 7.2: Q-RACH Approach

unable to get access. In such a case, delay critical user will suffer the most because of their delay constrained nature. To mitigate this congestion issue particularly for DC-UEs, a combined approach is proposed which is Q-learning based RACH called Q-RACH with dynamic priority.

Fig.7.2 shows how Q-learning is applied in our work, the main RACH frame has a maximum of 10 Random Access Opportunities (RAO) and is considered as an initial global frame that will repeat after every 10ms. Firstly, in the global frame we have separated the resources for DC and NDC UEs, so that there will be no collision between these two user groups. After the separation, Q-learning and SA-RACH are implemented in the global RACH frame, where both the protocol (SA-RACH and Q-RACH) will work individually. A Q-learning assisted user use only the specific slot for RACH operation. In Fig.7.2, filled slots marked “M” are denoted as Q-slots, which will be used for learning. On the other hand, the white slots marked as “H” will be used as SA-RACH. We have used separate resources for DC and NDC in a priority or emergency scenario, this two-step priority will be implemented by using our previous model which is a coded preamble based PD-RACH scheme. The algorithm for coded preamble based dynamic RACH is given in Algorithm-2, and Fig.6.5 (chapter 6) shows a significant improvement in delay by handling various RACH parameters dynamically in our previous study. The basic of Q-learning is described in the later section.

7.2.2 Coded Preamble based Dynamic Q-RACH

A RA process can be controlled by eNB by restricting or allocating RACH resources and also by changing RACH parameters only if the eNB has information about the current network condition. The available information to an eNB is limited to the number of available preambles, the total number of successful or collide preambles and the number of registered devices. Although an eNB has knowledge about the total number of UEs in a network, it is extremely difficult to manage the RA process due to massive access requests in a short period of time, and it is even more difficult when H2H and M2M are combined in a shared RACH process. In this work, a single preamble RACH is considered with preamble coding technique, where preamble code=0 for the non-priority non-delay critical user group (NDC-UEs) and preamble code=1 for the delay-critical priority user group (DC-UEs). The main objective of this preamble coding is to classify different user groups so that priority can be given to them. A frame-based statistic strategy is adopted (described in Algorithm 1) with a frame-based Random Access cycle, RA_{cycle} and $RA_{cycle} <$ SIB2 Broadcast period. RAO per frame defined by $PrachConfigIndex$ which is known and broadcast via eNB in every 80ms. Frame duration for LTE is 10ms. In our work, RAcycle =40ms and SIB 2= 80ms. Total RAO in RA_{cycle} can be calculated. In our work, we use a coded preamble technique to distinguish between H2H and M2M preamble. After every RA_{cycle}, eNB will calculate the total number of RACH requests and the total number of successful RACH by counting the RAR messages. After each RA_{cycle} preamble collision probability for overall UEs and for delay-critical UEs are calculated using (7.1) and (7.2).

$$P_C = \frac{total\,preamble\,collision\,in\,RA_{cycle}\,(overall)}{Total\,RAO\,in\,RA_{cycle}}, \tag{7.1}$$

where, P_C= Preamble collision probability (overall)

$$\text{Total preamble collision in} RA_{cycle} = \text{No. of total attempts} \\ - \text{No.of successful RACH}$$

$$\text{Total RAO in} RA_{cycle} = \frac{RA_{cycle}}{\text{Frame duration}} * \text{RAO per frame}$$

Also, preamble collision probability for delay-critical UEs (DC-UEs) in Random Access cycle time RA_{cycle} can be calculated using the coded preamble technique.

$$P_{C_{DC}} = \frac{total\,preamble\,collision\,in\,RA_{cycle}(DC-UE)}{Total\,RAO\,in\,RA_{cycle}} \tag{7.2}$$

where, $P_{C_{DC}}$= Preamble collision probability (DC-UE)

These preamble collision values (P_C and $P_{C_{DC}}$) are compared with predefined preamble collision thresholds ($Pc_{threshold(min)}$, $Pc_{threshold(max)}$ and $P_{DC_{threshold}}$) in order to provide a two-stage prioritisation. A priority flag is used to enable these stages where first stage is called priority without emergency stage and the other stage is defined as priority with emergency stage. In case when both Priority and Emergency flags are set to FALSE, both delay critical and non-delay critical user groups share the same RACH parameters with equal rights. On the other hand, when only the Priority flag is set to TRUE which indicates first stage priority mode for DC-UEs and to provide priority high $PrachConfigIndex$ will be assigned to DC-UEs as well as a frame-based backoff will be assigned instead of slot based backoff for NDC-UEs. In the emergency state in order to minimise the collision probability, we restrict NDC-UEs by limiting the maximum retransmission number (no backlogged request for NDC-UEs will be accepted only newly generated requests will be accepted). Another case is when both the Priority and Emergency flag will set to TRUE and this stage is called the Priority-with Emergency stage. In this case, to avoid collision among delay-critical user groups in order to minimise delay a Q-learning based unique slot selection technique is applied which is described in the following section. Finally, eNB will set group-based specific RACH configuration parameters ($PrachConfigIndex$, Max retransmission value, backoff period) based on flag indicator and will broadcast via System information block 2 (SIB2). Full details of the coded-preamble based Dynamic RACH scheme can be found in chapter 6, section 6.6.2.

Algorithm 2 Collision Probability and Preamble coded based Dynamic RACH Prioritisation

eNodeB side:
Initialisation:
T $\leftarrow 0, RA_{\text{cycle}}$
Set initial RACH Paremeter:
All Flag = FALSE, Max Re-TX value, $PrachConfigIndex$
eNB Broadcast initial RACH Paremeter via SIB2
for $T \leftarrow 0$ to T **do**
  if $T == RA_{\text{cycle}}$ **then**
    Calculate P_C and $P_{C_{DC}}$ via eqn. (7.1 and 7.2)
    UpdateRA_{cycle}
    if $P_C < Pc_{threshold(min)}$ **then**
      Priority Flag = FALSE
      Emergency Flag = FALSE
    end if
  else if $Pc_{threshold(max)} > P_C > Pc_{threshold(min)}$ **then**
    Priority Flag = TRUE
    if $P_{C_{DC}} < P_{DC_{threshold}}$ **then**
      Emergency Flag = FALSE
    else
      Emergency Flag = TRUE
    end if
  else if $P_C > Pc_{threshold(max)}$ **then**
    Priority Flag = TRUE
    Emergency Flag = TRUE
  end if
  if $T ==$ eNB Broadcast time **then**
    Set Flag & Broadcast SIB2
  end if
end for

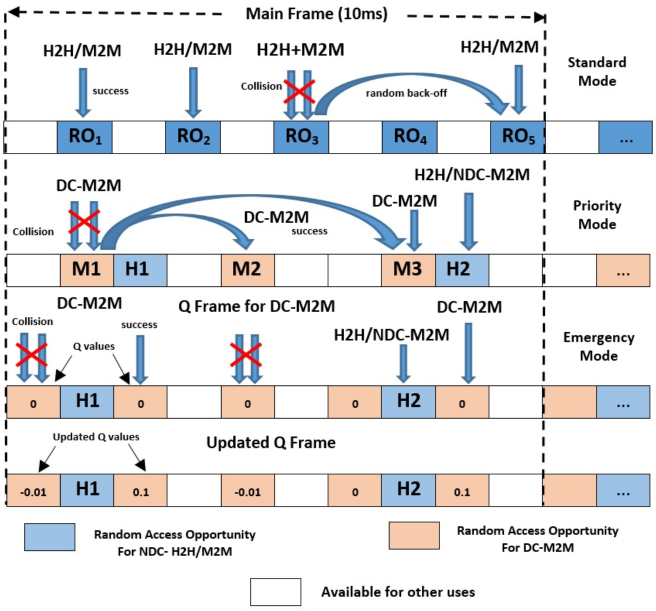

Figure 7.3: Q-Learning assisted Prioritised Dynamic RACH

7.2.3 Q-Learning assisted Prioritised Dynamic RACH (QPDR)

In this method, a slotted Aloha Q-learning scheme is adopted (described in Algorithm 2) to prioritise DC-M2M users only, where other users (H2H/NDC-M2M) are using conventional SA-RACH. Q learning is a basic model of reinforcement learning with the trail-error-and technique, which allows early system convergence. The idea behind this technique is to avoid collision by learning and acquiring a dedicated slot during the contention process as stated in [72]. In [72] a combined RACH access scheme is proposed to control M2M traffic in order to reduce its impact on a cellular network. The QL-RACH access scheme uses an intelligent slot assignment strategy in order to avoid collisions amongst the M2M users. In our work, we use Q-learning for prioritisation which aims at solving the delay issue for a specific user group called the delay-critical user group.

Fig.7.3 shows how Q-learning assisted Prioritised Dynamic RACH works, where a global frame is considered which has 10 slots. In the Standard mode, RO is Random Access Opportunity for all the UEs, where a UE chooses a RA slot randomly and if two or more UE select the same slot collision occurs and after a random backoff a UE can retransmit. On the other hand, in the Priority mode, the slots coloured in blue are for H2H/NDC UEs and the peach coloured boxes are for DC UEs only. If a UE experienced collision they only can retransmit on their specific slots. Priority mode separates the UEs in different RACH resources in order to minimise collision. After SIB2 reception if the UE finds the Emergency Flag is TRUE which means delay-critical UEs needs a priority with an emergency. To implement Q-learning, in this case, DC-UEs will use different $PrachConfigIndex$ than

Algorithm 3 Q-Learning assisted Prioritised Dynamic RACH (QPDR) Configuration for DC-UEs

After Successfully SIB2 reception:
if Emergency Flag = TRUE **then**
  Set Emergency RACH values for DC-M2M
  Initialisation Q-learning values:
  $Q \leftarrow 0, r\ \alpha$
  Input : where, Q is the current Q-value
  r is the reward or punishment, α is the learning rate
  Calculate Q' with updated Q value via eqn. (7.3)
end if
if Priority Flag = TRUE **then**
  Set Priority RACH values for DC-M2M
end if
if Priority Flag = FALSE **then**
  Set Standard RACH values for M2M and H2H
end if

NDC-UEs to make M2M Q-Frame from the mainframe as shown in Fig 7.3. Using different $PrachConfigIndex$ will separate the DC and NDC-UEs in the first place and will avoid collision among them. The DC-UEs number is set equal to the number of RAO in a Q-Frame to get the optimum learning. Firstly, all the Q-values are initialised to zero and Q-values are used to keep the transmission history in each slot. Every Q-Frame will update with every mainframe and the slot value will update at every RACH access attempt. Based on every successful or failed transmission Q-values will update using the following model:

$$Q' = (1-\alpha)\ Q + \alpha r \tag{7.3}$$

Where Q is the current Q value, α is the learning rate= 0.01 and r is the reward (+1) or punishment (-1).

Let's assume a UE randomly select a slot from the Q-Frame and initially all the Q-values are the same. If the transmission is successful (which means positive reward received) and using (7.3) the Q value of the slot is updated. In the next Q-Frame a UE before sending a request instead of selecting a random slot, UE will send a request in the slot with the highest Q value. If there are multiple slots with the same highest Q value, one is selected at random. By doing this error-and-trail learning, all the DC-UEs can find a unique slot with the highest value, which avoids interfering with other UEs.

Table 7.1 represents the parameters used in this simulation based on LTE standards.

Table 7.1: Simulation parameters for QPD-RACH

Parameter	**Value**
RA Slot period	1ms
Frame Duration	10ms
Retransmission limit	5
PRACH configuration index	7,9,10,12,13
Number of Preamble sequence	1
Preamble format	1
RAO per frame	5,7
RAR Window size	5ms
Back-off period	10/20 slots
Total No. of UEs	100
No. of DC-M2M user per frame	5
Preamble collision threshold (overall) $Pc_{threshold(min)}$ and $Pc_{threshold(max)}$	0.25 and 0.4
Preamble collision threshold for DC-UEs $PDC_{threshold}$	0.25
Random Access cycle (RA_{cycle})	40ms
Learning rate	0.01

7.3 Performance Evaluation of Proposed QPD-RACH

7.3.1 Simulation Scenario and parameters

Riverbed Modeller is used as simulator to evaluate the performance of combined Proposed QPD-RACH and SA-RACH schemes with considering delay and non-delay UEs. We also consider some assumptions for both UEs to represent the scenario that allows us to analyse RACH of conventional slotted-ALOHA according to the standards. These assumptions are:

- All packets length is equal to the length of the slot.
- RACH request arrival follows the Poisson arrival process.
- The system is perfectly synchronised with every user and can transmit only at the beginning of a slot.
- All users share a single RACH (one preamble available).

Based on the assumptions the simulation is modeled with the introduction of the RACH request generation from each user group (H2H and M2M) following Poisson distribution having a mean inter-arrival time τ_{ia} determined by the traffic load G as shown below.

$$\tau_{ia} = P_{len}N/GT_{rate} \tag{7.4}$$

Where P_{len} is the preamble length/packet length in bits, N is the total user in the system, G is the traffic load in Erlangs and T_{rate} is the transmission rate in bits/sec.

The collaboration between the user groups (H2H & M2M) is controlled by providing individual load in each group. We used a fixed load for M2M traffic at 0.1E and a varying load for H2H traffic from 0.02E to 0.2E to achive slotted-ALOHA channel capacity. Therefore the total generated traffic in the system using the load sharing is expressed as:

$$G_{Total} = G_{H2H} + G_{M2M} \tag{7.5}$$

Where G_{Total} is the total generated traffic and G_{H2H} and G_{M2M} is traffic generated by H2H and M2M respectively.

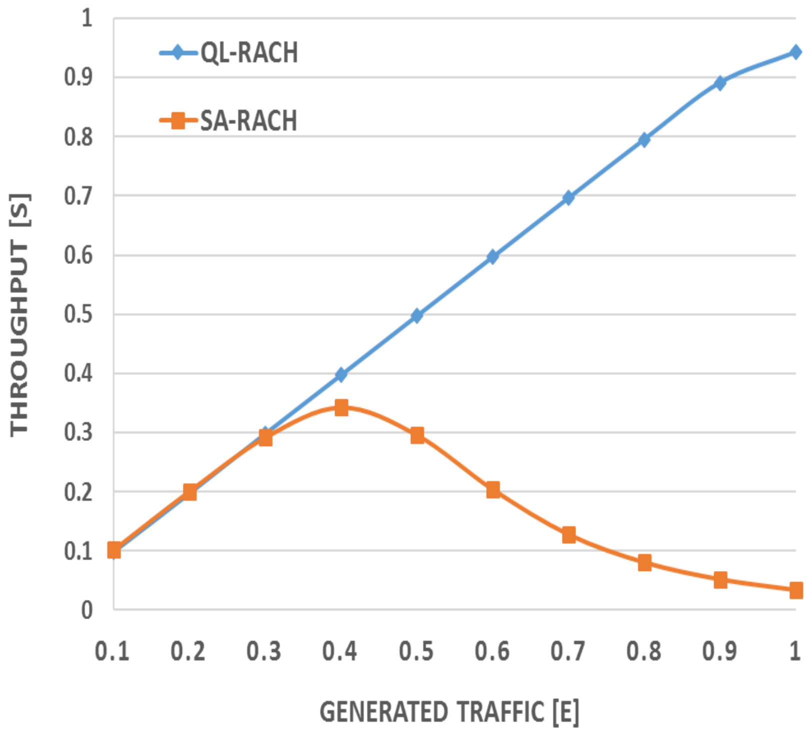

Figure 7.4: Throughput vs. generated traffic.

7.3.2 Simulation results and Discussion

To investigate the effectiveness of Q-learning, firstly, we compare the single user group RACH-throughput performance of the SA-RACH with retransmission scheme and the steady-state RACH-throughput of the QL-RACH scheme. From Fig.7.4, it can be seen that the SA-RACH throughput increases with the increase in the generated traffic. However, immediately after the channel throughput limit (36% which is the maximum capacity of the s-ALOHA) is reached, the aggregated traffic increases to the point that the s-ALOHA scheme can no longer support the traffic. This is why we can notice the throughput dropping with an increase in the traffic, to the extent that the channel becomes unstable. On the other hand, with steady-state of the Q-learning, the QL-RACH scheme offers up to 100% throughput. This is because there are no collisions since the scheme is contention free. In Fig.7.4, the convergence is achieved because we have considered that the number of UEs are equal to the number of Q-slots available in a Q-frame. On the contrary, if the number of UEs increases beyond the number of Q-slots, the convergence will not be achieved because the system will never be contention free. If multiple UEs select the same Q-slot at the same time, collision will occur and punishment will be given to both UEs, hence the Q-value will degrade and the system will not converge.

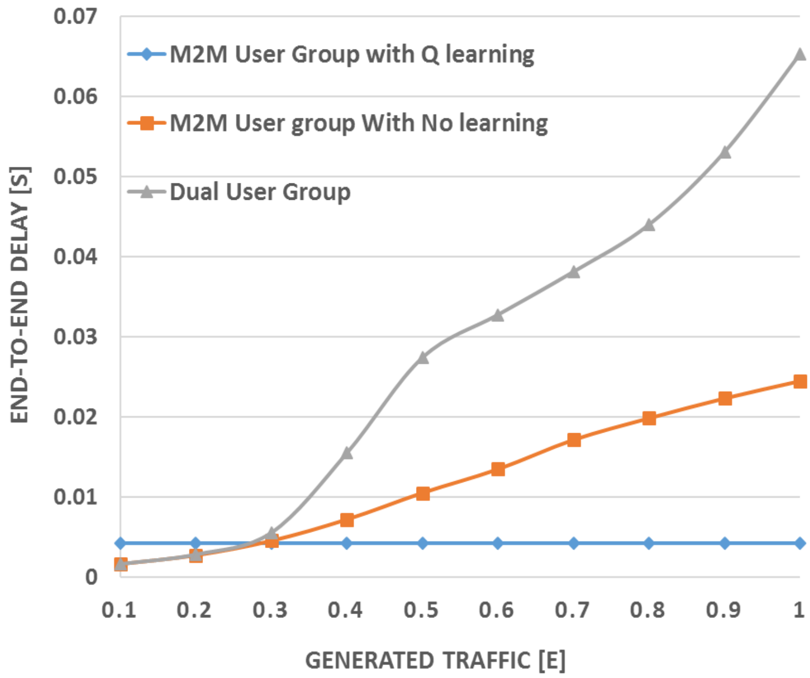

Figure 7.5: End-to-End delay vs. generated traffic.

Fig. 7.5 shows the delay performance for different user groups, where Dual User groups include both M2M and H2H. Both QL-RACH and SA-RACH is considered where M2M user group use Q-learning, and on the other hand, H2H user group use conventional SA-RACH. Here generated traffic is considered as an aggregated traffic for both H2H and M2M. Fig.7.5 shows that in low traffic scenarios both user group perform well with very low delay, on the contrary, with the increasing load the delay for M2M with no learning as well as dual user group increased. It can be seen that using QL technique shows a minimum delay profile with respect to non-learning and dual user group scenario. By giving priority to the M2M user group in a convergence mode when all M2M have a specific slot and there will be no collision between them, it will become contention free, hence the delay is less compared to the non-learning mode. The only issue with Q-learning is, it will not achieve convergence if the number of devices active in frame is greater than the Q-slot available in that specific frame. In our work, we have assumed that the number of active devices is equal to the Q-slot value.

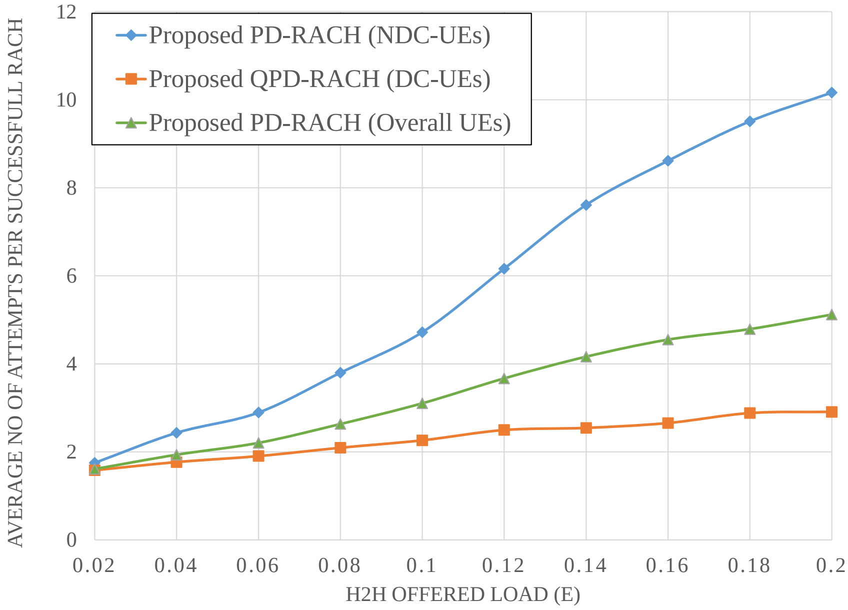

Figure 7.6: Average attempts per successful RACH vs. H2H offered load.

7.3.3 Dual User Group

In this work, to evaluate the performance well within the slotted-ALOHA channel capacity, we vary the H2H offered load up to 0.2E. In this dynamic technique a higher and a lower bound is considered based on the priority and emergency state. In priority mode, the higher $PrachConfigIndex$ value will be assigned to the priority user and in an emergency mode, the slot will be divided so the priority user can access the Q-learning mode and use the RACH global frame as Q-Frame. In convergence mode, all the DC-UE will have their own slot, and the collision will resolve among the DC-UEs, also using different $PrachConfigIndex$ there will be no collision among NDC and DC UEs. We have considered two different modes: priority and emergency with priority modes. The eNB can detect these two modes dynamically by using a coded preamble based collision probability model. As shown in Fig. 7.6 with the increasing H2H load the average number of attempts per successful RACH also increases, due to the resource limitation and massive access request. When two or more UEs select the same preamble or select the same resource slot to send the RACH request, collision occur and both UE can retransmit the request after a random interval. When the number of UEs increases drastically the collision also increases, as a result, UEs need to try more times than usual and the average number of attempts per successful RACH increases. In eNB after every 40ms which is 4 global frame time, eNB checks the collision probability and uses Algorithm-1 to detect the emergency mode and response via SIB-2 with resource and priority information to all the UEs. If the Emergency flag is set to TRUE then Algorithm-3 activate

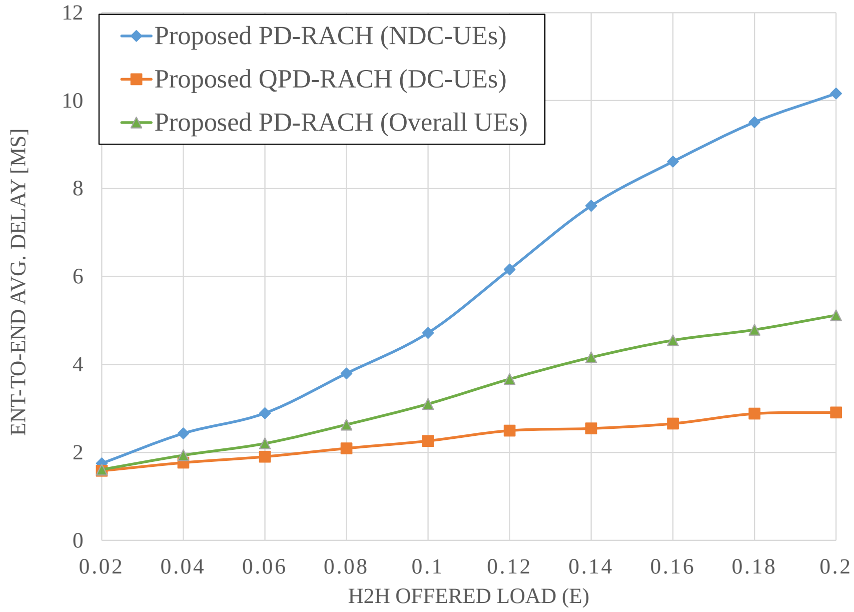

Figure 7.7: End-to-end delay vs. H2H offered load.

which is the Q-learning approach for M2M DC priority user. Q-learning plays a vital role to minimise the collision effect by providing a dedicated slot, which keeps the overall attempts for all user within the limit. Particularly in the convergence mode, all the DC-UEs will have a dedicated slot and can access the RACH channel without collision.

Fig.7.6 shows the average number of attempts per successful RACH access versus H2H offered load. After we investigate Q-RACH single user mode, where only M2M users both delay critical and non-delay critical exist, this investigation is based on dual user mode, where both H2H and M2M users co-exist. It shows that when the offered load for both H2H and M2M is equal to or less than 0.1E, the average number of attempts per successful RACH access is at a minimum, whereas this number increases dramatically with increasing load value. On the other hand, only DC user have a minimum effect on H2H rising load, our proposed scheme dynamically give priority by applying the Q-learning to the DC-UEs, even if in a high load scenario. Also, in overloaded case, the Q-learning play a vital role to minimise the collision effect by providing a dedicated slot, which keeps the overall attempts for all user within the limit.

Fig.7.7 shows the end-to-end delay with respect to H2H offered load. In this work, we have considered that the number of M2M UEs is fixed and the number of maximum UEs is equal to the number of maximum available RACH slots per frame. We have varied the H2H offered load to see the impact, as we can see with the increasing offered load of H2H, the end-to-end delay is also increasing and in high load scenarios, the dealy is very high

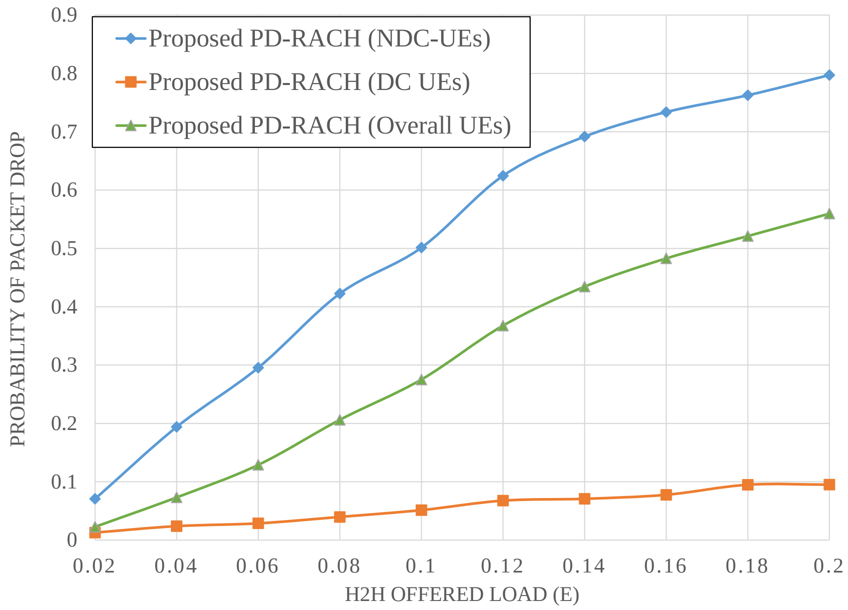

Figure 7.8: Probability of packet drop vs. H2H offered load.

for NDC-UEs. On the other hand, the performance of DC-UEs outperforms the other user group and it shows a minimum delay even with the high H2H load. This is because our proposed scheme plays a vital role in minimising the collision in a massive access scenario by applying Q-learning to the DC-UEs which provides dedicated slots in convergence mode. When DC-UEs get a dedicated slot to send the RACH request, it avoids the collision with H2H as well as within the same group. The average number of attempts per successful RACH is very low which results in a minimum delay. The results show that using a learning-based dynamic prioritisation technique can provide a QoS requirement based service to a specific user and the scheme is able to keep the delay lower than the conventional scheme.

Fig. 7.8 represent the packet dropping probability versus H2H offered load. In this work, to evaluate the performance well within the slotted-ALOHA channel capacity, we vary the H2H offered load up to 0.2E. As we can see from Fig. 7.8, at low load conditions both the DC and NDC-UEs perform well and the packet dropping probability is low, but under the heavier loads, the performance is poorer, almost 80% of packets are dropped when H2H offered load is 0.2E. This is because the increased H2H load and corresponding higher overall load result in a significant number of collisions due to the limited number of random access opportunities available in the frame. On the contrary, using our proposed learning-based scheme for DC-UEs shows a satisfactory performance by keeping the dropping probability very low in both low and high load conditions. When the system enters the emergency mode eNB informs UEs via SIB-2 and all the UEs can dynamically change the RACH parameters

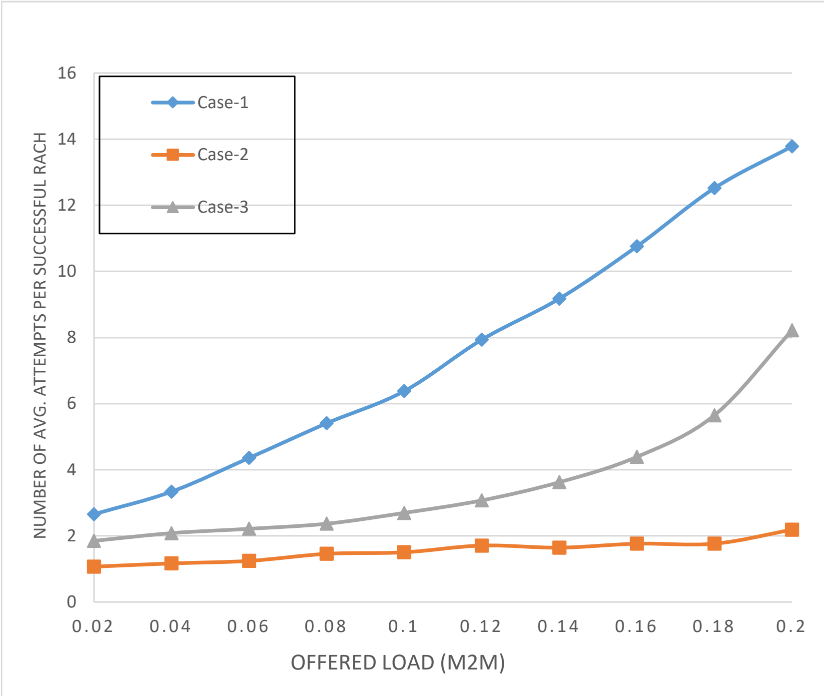

Figure 7.9: Impact of the number of M2M-UEs on Q-learning

and DC-UEs start using the Q-learning approach in order to achieve the priority access, where the collision rate minimises as a result the packet dropping probability also decreases. Also, in the convergence mode, all the DC-UEs has a dedicated RACH slot for the RACH access request and the system almost work as a non-contention based RACH procedure. From the figure, we can see that the packet dropping probability is very low and also maintain the value throughout the simulation by adjusting the parameters dynamically using our proposed learning assisted dynamic prioritised RACH.

In our work, Q-learning has been used to acquire dedicated slots for collision-free RACH in an emergency scenario to provide priority over the non-priority group of users. The Q-learning scheme is implemented by designing an M2M frame in which Q-slots are fixed size equal to the number of active M2M users in the system in order to achieve maximum performance as described in section 7.2.3. From Fig. 7.4, we can see that convergence is only possible if the number of UEs is equal to the number of Q-slots. However in some cases, the scenario could be different. In this section, in order to examine the impact of the number of M2M on Q-learning, we have considered 3 different cases,

- Case-1: The number of M2M UES is more than double the number of Q-slots
- Case-2: The number of M2M UEs is equal to the number of Q-slots
- Case-3: The number of M2M UEs is double the number of Q-slots

To evaluate these cases, we have considered M2M with Q-learning, and have evaluated the average number of attempts per slot value. We can see that in case-1 the average number of attempts is very high because the number of M2M UES is more than double the number of Q-slots, and the system is not in convergence mode. In a low load scenario, when the number of RACH requests is low the average no of attempts is also low with Q-learning but in a high load scenario, case-1 have the worst performance. On the other hand, in case-2 the average number of attempts is very low in both low and high load scenarios. All the UEs have their dedicated slot and no collision in the convergence mode is achieved. Also, in case-3, when we have doubled the value in a frame the convergence becomes unstable and it shows that at low load it still performs better but in the high load condition it becomes unstable. In our work, we have considered this case which is a limitation of our work but this issue can be solved by using a dynamic M2M-frame size, in which the Q-slot will be dynamically adjusted in order to maintain the convergence mode, which we have considered as a part of our future work.

7.4 Summary

We have presented a performance evaluation of contention-based RACH access, where delay-critical and non-delay-critical devices coexist. A Q-learning assisted coded preamble based prioritised dynamic RACH is proposed where a group-based separate Q-Learning technique is used for delay-sensitive M2M UEs. The idea behind this work is to use the Q-learning approach in an emergency scenarios caused by a massive number of access requests on RACH. The proposed scheme provides a two-step prioritisation mechanism that includes a combination of dynamic resource allocation of RACH and a specific Q-learning approach for collision avoidance between the same user groups. In an emergency case, Q-learning plays a vital role to minimise the collision by providing a dedicated slot, which keeps the overall attempts for all users within the limit. The simulation results show a positive impact on delay in highly loaded traffic scenarios by dynamically adjusting the RACH parameter. Using the proposed learning approach could greatly decrease access delay and a significant gain in the number of successful attempts for delay-critical UEs. For delay-critical UEs, our proposed algorithm outperforms the conventional scheme and shows that even as the H2H load is increased the learning-based prioritised dynamic RACH scheme is able to keep the delay lower than the conventional scheme.

Chapter 8

Conclusion and Future Work

8.1 Summary and Conclusion

The upcoming cellular networks require to be designed in such a way that can provision the massive number of MTC devices fulfilling their various QoS requirements along with enriching the access latency, scalability and network throughput. As per the prediction of 3GPP and other organisation members, simultaneous access attempts of the massive number of MTC devices will result in congestion of the RACH. This work explores the situation inflicted upon the cellular network with the inclusion of M2M communication. In chapter 2, we have provided a general overview of the machine to machine communications including applications, requirements and characteristics. Also, we have mentioned the challenges and advantages of cellular networks for M2M communications. Some potential enablers for M2M communication are also provided in the chapter in addition to the existing solution. As an emerging solution, ML techniques with their type and advantages are discussed. Taking into consideration the challenges and potential enablers for MTC in next-generation cellular IoT network, in this work we have focused on several challenges such as how to support a massive number of MTC devices, ensure a low latency based communication as well as how to solve the RAN congestion problem. To understand the RAN congestion problem in the RACH procedure we have provided a detailed discussion on RACH limitation and overload control mechanism in Chapter 3. Chapter 3, provides background information on LTE cellular network including its network structure, frame structure, and channel configuration. The chapter also reviews Random Access Channel, providing the RA procedures, the impact of M2M communication on RACH and the limitation on RACH. Then a literature review on the RACH overload control mechanism is given in detail including both 3GPP and non-3GPP based solutions. Also, as an emerging solution learning-based technique is also discussed. An overview of the communication simulation system has been discussed in chapter 4. Riverbed Modeller is used as a simulation tool in this study is described and a reference model of the developed simulator as well as simulator design steps are also presented. The key metrics used to assess the performance of the simulated protocols i.e. throughput, blocking probability, and end-to-end delay of the network are also included. Finally, the traffic model used in this thesis is discussed. In chapter 5, the introduction of the basic two ALOHA schemes (slotted and pure) with their performance comparison is presented. Then investigation on two-step slotted-ALOHA RACH with M2M traffic which coexists with H2H traffic is provided by evaluating performance using throughput, delay and blocking probability. Finally, a novel group based two step slotted-ALOHA is introduced in this chapter. A dynamic RACH configuration approach for LTE is used to reduce the massive access problem and the access delay of a specific user group called M2M DC-UE with high priority without affecting the total number of accesses. Based on the simulations results, we observe that this approach reduces the access delay of M2M devices with high priority. Chapter 6, introduced Priority Based Random Access which enables M2M users (specially Delay critical UEs) to coexist with H2H users even in a massively loaded scenario. A coded preamble based collision probability model is used to define the priority state where DC-UEs are given the highest priority. This work considered a two-stage priority model, one called priority state and another is emergency state. The simulation results show that even in a highly loaded traffic scenario, by using this scheme a DC-UE can dynamically change its RACH parameter values given via SIB2 in order to get access in an emergency or overloaded situation. A significant decrease in access delay and a significant gain in the number of successful attempts is observed through the simulation results. Additionally, we explore the benefits of machine learning as an effort to minimise the RACH overload/congestion of the cellular network. A novel coded preamble based dynamic Q-leaning assisted RACH is proposed for further prioritisation in order to serve priority/required QoS to the priority user groups and simulation results show a positive impact on delay.

8.1.1 Original Contributions

Two-step Group-based RACH: A two-step group based RACH is proposed in our work, where each group have their own different requirements. In order to provide the different QoS requirements instead of having a common RACH, a group based RACH can efficiently handle the group. In our work, we have considered two specific groups: one is called delay-critical and the other is called non-delay-critical. As a novel contribution, a two-step RACH in LTE is investigated and a group-based as well as Separate Resource Allocation based RA scheme is proposed.

Prioritised Dynamic RACH (PD-RACH): We proposed a novel RACH scheme called Prioritised Dynamic RACH (PD-RACH). This novel proposed RACH mechanism considers a shared RACH Channel with the coexistence of H2H and M2M UEs. Also, the M2M users are categorised into two groups: delay-critical (DC-UEs) and non-delay-critical (NDC-UEs). We propose a Priority-based slotted-ALOHA Dynamic RACH (PD-RACH) scheme to control the interaction and the delay for the specific delay-critical user group, whereas in the traditional slotted-ALOHA RACH an equal opportunity based fixed RACH is used for both H2H and M2M users. The proposed scheme provides a two-step prioritisation mechanism and includes a combination of dynamic allocation of RACH resources and a group based specific backoff by utilising random access cycle-based information. The eNB can monitor this information to adjust RA resources and other parameters like backoff and max-retransmission value in order to minimise delay and provide priority to delay-critical UEs over non-delay-critical UEs. Moreover, we analyse our proposed scheme and simulations results validate the effectiveness of the proposed scheme compared to the conventional scheme.

Q-learning assisted prioratised RACH (QPD-RACH): We have studied a shared RACH Channel with the coexistence of H2H and M2M devices and also considered delay critical and non-delay critical UEs. A Q-learning based Prioritised Dynamic RACH (QPD-RACH) scheme is proposed in order to provide priority to the delay-critical user group (DC-UE) in a congestion scenario caused by massive MTC devices. To monitor and adjust RA resource parameters as well as to set the priority, eNB used a coded preamble based collision probability model which provides a two-step prioritisation scheme. In the first step, UE (both DC and NDC) adjust the RACH parameters like backoff value, $PrachConfigIndex$, Max-ReTx value and in the second step, a Q learning approach is applied to avoid collision among delay critical UEs in order to minimise delay and provide priority access to delay-critical UEs over non-delay-critical UEs. A Riverbed Modeller based simulation study for the contention-based RACH has been studied, and the results show that the proposed scheme can significantly decrease the access delay for the priority user group.

8.1.2 Hypothesis Revisited

"A Dynamic Prioritised Random Access strategy can provide priority access to a specified user group termed Delay-critical UEs, by minimising access delay and providing required Quality of Service (QoS) in a congested network scenario caused by enormous machine-type devices."

Machine-type communications (MTC) is considered as one of the promising technologies over the next decade and these MTC devices are continuously increasing. When a large number of MTC devices try to access a cellular base station simultaneously using the existing LTE/LTE-A Random Access Channel (RACH), the collision probability among Machine-to-Machine (M2M) and Human-to-Human (H2H) devices increase drastically. Thus the performance of RACH degrades sharply which causes significant blocking and huge access delay. Delay critical applications, cannot, therefore, be served using the legacy RACH protocol.

In our work, we have proposed several advanced RA schemes including priority for specific user groups called delay-critical UEs. Our proposed schemes show effectiveness on the conventional RACH scheme and provide priority to the specific user group. A group-based technique is proposed where users are group and assigned parameters based on their requirements. A coded preamble based collision probability model is used to define the priority state where DC-UEs are given the highest priority. This work considered a two-stage priority model, one called priority state and another is emergency-state. Also an emerging technology we have incorporated machine learning in our work, as an efficient and simple learning technique Q-learning based adopted in our work which shows the effectiveness of learning. The prioritised learning-based RACH minimises the delay for a specific user group and, the result shows some significant improvements in the adaptability of combined RACH schemes; hence, upholding the hypothesis of the thesis.

8.2 Future Work

Dual Q-Based Prioritised RACH: In our work, we have considered machine learning and adopted a Q-learning approach, in one of our proposed schemes called Q-learning assisted Prioritised RACH where we have used a single-mode Q-learning only for delay critical users. Looking at the performance of the proposed schemes in this thesis, it is clear that, the proposed scheme significantly improves the RACH-throughput performance of both H2H and M2M users especially compared with the SA-RACH scheme. As a further modification to our work, a two-stage Dual-Q learning approach can be considered which can provide further priority where priority will be crucial. As a part of our future plan, we will investigate Q learning approach for eNB so that it can handle a massive number of access requests with the help of a learning-based approach.

Preamble and Slot-separation based Prioritised RACH: In our work, we have considered a single channel RACH where a single preamble is considered for all users in a shared manner and no slot reservation technique is used. A multi preamble slot separation based prioritised RACH can be considered as part of future work, where preamble can be divided into specific user groups based on their requirement and priority mode. Also, slots can be separated and reserved for a specific user group (delay-critical UEs) by using a learning approach such as Q-learning based slot separation technique in order to provide priority access to the required UEs.

Q-learning assisted preamble separation for RACH Prioritisation: We have studied a shared RACH Channel with the coexistence of H2H and M2M devices and also considered delay critical and non-delay critical UEs. A Q-learning based Prioritised Dynamic RACH (QPD-RACH) scheme is proposed in order to provide priority to the delay-critical user group (DC-UE) in a congestion scenario caused by massive MTC devices. To monitor and adjust RA resource parameters as well as to set the priority, eNB used a coded preamble based collision probability model which provides a two-step prioritisation scheme. The results show that the proposed scheme can significantly decrease the access delay and they significantly improve the RACH-throughput performance for the priority user group especially compared with the SA-RACH scheme. Similarly, Q-learning can be applied in order to separate Preamble in multi preamble RACH. Using Q-learning, UE can select a dedicated slot as well as it can select a dedicated preamble from the preamble pool, which will mitigate the collision problem in conventional RACH and could provide priority access to the required UEs.

References

[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, 4th Quart., 2015.

[2] S. K. Sharma and X. Wang, "Toward Massive Machine Type Communications in Ultra-Dense Cellular IoT Networks: Current Issues and Machine Learning-Assisted Solutions," in IEEE Communications Surveys and Tutorials, vol. 22, no. 1, pp. 426-471, First quarter 2020, doi: 10.1109/COMST.2019.2916177.

[3] IMT Vision-Framework and Overall Objectives of the Future Development of IMT for 2020 and Beyond, ITU-R Standard M.2083-0, Sep. 2015.

[4] S. Lien, K. Chen, and Y. Lin, “Toward ubiquitous massive accesses in 3GPP machine-to-machine communications,” IEEE Commun. Mag., vol. 49, no. 4, pp. 66–74, Apr. 2011.

[5] H. Shariatmadari, R. Ratasuk, S. Iraji, A. Laya, T. Taleb, R. Jantti, ¨ A. Ghosh, Machine-type communications: current status and future perspectives to- ward 5g systems, IEEE Commun. Mag. 53 (9)(2015) 10–17

[6] K. Zheng, S. Ou, J. Alonso-Zarate, M. Dohler, F. Liu, and H. Zhu, “Challenges of massive access in highly dense LTE-advanced networks with machine-to-machine communications,” IEEE Wireless Commun., vol. 21, no. 3, pp. 12–18, Jun. 2014.

[7] You X H, Wang C-X, Huang J, et al. Towards 6G wireless communication networks: vision, enabling technologies, and new paradigm shifts. Sci China Inf Sci, 2021, 64(1): 110301, https://doi.org/10.1007/ s11432-020-2955-6

[8] 12 You X H, Yin H, Wu H Q. 6G and wide area Internet (in Chinese). Chinese J Int Things, 2020, 4: 12–18

[9] Z. Dawy, et al., “Toward massive machine type cellular communications,” IEEE Wireless Communications, vol. 24, no. 1, pp. 120–128, Feb. 2017.

[10] "Study on RAN Improvements for Machine-Type Communications", Sep. 2011.

[11] https://cdn.ihs.com/www/pdf/IoTn ebook.pdf

[12] Palattella, M. R., Dohler, M., Grieco, A., Rizzo, G., Torsner, J., Engel, T., and Ladid, L. (2016). Internet of things in the 5G era: Enablers, architecture, and business models. IEEE Journal on Selected Areas in Communications, 34(3), 510-527.

[13] Kalyani, Vijay and Sharma, Deeksha. (2015). IoT: Machine-to-Machine (M2M), Device-to-Device (D2D) Internet of Everything (IoE) and Human-to-Human (H2H): Future of Communication. Journal of Management Engineering and Information Technology. 2394-8124.

[14] ETSI TS 102 690 V1.1.1 , “Machine-to-Machine communications (M2M); Functional architecture,” Octr. 2011.

[15] Chen, M. et al. “Machine-to-Machine Communications: Architectures, Standards and Applications.” KSII Trans. Internet Inf. Syst. 6 (2012): 480-497.

[16] Ghavimi and H. H. Chen, “M2M communications in 3GPP LTE/LTE-A networks: Architectures, service requirements, challenges, and applications,” IEEE Commun. Surveys Tuts., vol. 17, no. 2, pp. 525–549, Secondquarter 2015.

[17] A. Laya, L. Alonso, and J. Alonso-Zarate, "Is the Random Access Channel of LTE and LTE-A Suitable for M2M Communications? A Survey of Alternatives, "Communications Surveys and Tutorials, IEEE, vol. 16, pp. 4-16, 2014.

[18] A. Lo, Y. W. Law, M. Jacobsson, and M. Kucharzak, ”Enhanced LTEadvanced random-access mechanism for massive machine-to-machine M2M) communications,” in in 27th World Wireless Research Forum (WWF) Meeting, 2011.

[19] 3GPP TS 22.368 v11.2.0 ”service requirements for Machine-Type Communications,” Jun. 2011.

[20] Juniper networks, "machine-to-machine (M2M)-the rise of the machines", White paper, 2011.

[21] https://blog.technavio.com/blog/iot-security-concerns-leading-to-massive-market-growth.

[22] Verma, P.K., Verma, R., Prakash, A., Agrawal, A., Naik, K., Tripathi, R., Alsabaan, M., Khalifa, T., Abdelkader, T. and Abogharaf, A., 2016. Machine-to-Machine (M2M) communications: A survey. Journal of Network and Computer Applications, 66, pp.83-105.

[23] Y. Saleem, N. Crespi, M. H. Rehmani, and R. Copeland, “Internet of Things-aided smart grid: Technologies, architectures, applications, prototypes, and future research directions,” CoRR, vol. abs/1704.08977, Apr. 2017.

[24] K. Zheng, F. Hu, W. Wang, W. Xiang, and M. Dohler, “Radio resource allocation in LTE-advanced cellular networks with M2M communications,” IEEE Commun. Mag., vol. 50, no. 7, pp. 184–192, Jul. 2012.

[25] M. Bkassiny, Y. Li, S.K. Jayaweera, A survey on machine-learning techniques in cognitive radios, IEEE Commun. Surv. Tutor. 15 (3) (2013) 1136–1159, doi: 10. 1109/SURV.2012.10 0412.0 0 017

[26] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.C. Chen, L. Hanzo, Machine learning paradigms for next-generation wireless networks, IEEE Wirel. Commun. 24 (2) (2017) 98–105, doi: 10.1109/MWC.2016.1500356WC .

[27] M. Chen, U. Challita, W. Saad, C. Yin, M. Debbah, Machine learning for wire- less networks with artificial intelligence: a tutorial on neural networks, 2017 . arXiv: 1710.02913 .

[28] L. R. Izquierdo and S. S. Izquierdo, "Dynamics of the Bush-Mosteller learning algorithm in 2x2 games," 2008.

[29] R. Li, , “Intelligent et al., 5G: When cellular networks meet artificial intelligence,” IEEE Wireless Commun., vol. 24, no. 5, pp. 175–183, Oct. 2017.

[30] X. Wang, X. Li, and V. C. M. Leung, “Artificial intelligence-based techniques for emerging heterogeneous network: State of the arts, opportunities, and challenges,” IEEE Access, vol. 3, pp. 1379–1391, 2015.

[31] N. Kato, et al., “The deep learning vision for heterogeneous network traffic control: Proposal, challenges, and future perspective,” IEEE Wireless Commun., vol. 24, no. 3, pp. 146–153,June 2017.55. A-15, 16

[32] A. Imran, A. Zoha, and A. Abu-Dayya, “Challenges in 5G: how to empower SoN with big data for enabling 5G,” IEEE Network, vol. 28, no. 6, pp. 27–33, Nov 2014.

[33] Sharma, Shree Krishna, and Xianbin Wang. "Towards Massive Machine Type Communications in Ultra-Dense Cellular IoT Networks: Current Issues and Machine Learning-Assisted Solutions." arXiv preprint arXiv:1808.02924 (2018).

[34] Richard S. Sutton and A. G. Barto, Reinforcement learning: An introduction, Cambridge MA: MIT Press, 1998.

[35] “Study on RAN improvements for machine-type communications,” 3GPP, Sophia Antipolis, France, Rep. TR 37.868, 2012.

[36] J. Choi, “On the adaptive determination of the number of preambles in RACH for MTC,” IEEE Commun. Lett., vol. 20, no. 7, pp. 1385–1388, Jul. 2016.

[37] Z. Li, J. Chen, R. Ni, S. Chen, X. Li, and Q. Zhao, “Enabling heterogeneous MMTC by energy-efficient and connectivity-aware clustering and routing,” in Proc. IEEE Globecom Workshops (GC Wkshps), Dec. 2017, pp. 1–6.

[38] A. Samir, M. M. Elmesalawy, A. S. E.-D. Ali, and I. Ali, “Partial contention free random access protocol for M2M communications in LTE networks,” J. Wireless Netw. Commun., vol. 6, no. 3, pp. 66–72, 2016.

[39] I. M. Delgado-Luque et al., “Evaluation of latency-aware scheduling techniques for M2M traffic over LTE,” in Proc. 20th Eur. Signal Process. Conf. (EUSIPCO), Bucharest, Romania, 2012, pp. 989–993.

[40] S. Ali, N. Rajatheva, and W. Saad, “Fast uplink grant for machine type communications: Challenges and opportunities,” CoRR, vol. abs/1801.04953, Jan. 2018. [Online]. Available: http://arxiv.org/abs/1801.04953

[41] 3GPP TS 36.300 V13.2.0, “Evolved Universal Terrestrial Radio Access (EUTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2,” Jan. 2016.

[42] L. R. Marcus, A. T. Michael, F. M. Robert, and A. R. Richard, "Evolution of the air interface of cellular communications systems toward 4G realization," Communications Surveys and Tutorials, IEEE, vol. 8, pp. 2-23, 2006.

[43] A. Larmo, M. Lindstrom, M. Meyer, G. Pelletier, J. Torsner, and H. Wiemann, "The LTE link-layer design," IEEE Communications Magazine, vol. 47, pp. 52-59, 2009.

[44] Understanding LTE (http:www.sharetechnote.comDocsanritsuunderstandinglte6.pdf)accessed September 2014.

[45] J. Zyren and W. McCoy, "Overview of the 3GPP long term evolution physical layer," Freescale Semiconductor, Inc., white paper, 2007.

[46] Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation, Dec. 2011.

[47] M. Amirijoo, P. Frenger, F. Gunnarsson, J. Moe, and K. Zetterberg, "On self-optimization of the random access procedure in 3G long term evolution,"in IEEE International Symposium on Integrated Network Management-Workshops, 2009. (IM'09), , pp. 177-184, 2009.

[48] 3GPP TS 36.321 V10.0.0, "Evolved universal terrestrial radio access (E-UTRA) medium access control (MAC) protocol specification," Dec. 2010.

[49] D. Katselis, "Some Preamble Design Aspects in CP-OFDM Systems," in IEEE Communications Letters, vol. 16, no 3, pp. 356-359, March 2012.

[50] 3GPP TS 36.321 V12.7.0, "Evolved Universal Terrestrial Radio Access: Medium Access Control," France, Sep. 2015.

[51] X. Yang, A. Fapojuwo, E. Egbogah, "Performance analysis and parameter optimization of random access backoff algorithm in LTE", Proc. IEEE Veh. Technol. Conf., pp. 1-5, Sep. 2012.

[52] 3GPP TR 37.868 V0.7.0, "Study on RAN improvements for machine type communications," Oct. 2010.

[53] Further Performance Evaluation of EAB Information Update Mechanisms, Germany:Intel, Feb. 2012.

[54] Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC) Protocol Specification, Dec. 2015.

[55] S. Duan, V. Shah-Mansouri, and V. W. S. Wong, "Dynamic access class barring for M2M communications in LTE networks," in Proc. IEEE Global Commun. Conf., Dec. 2013, pp. 4747–4752.

[56] J. P. Cheng, C. h. Lee, and T. M. Lin, "Prioritized random access with dynamic access barring for RAN overload in 3GPP LTE-A networks," in Proc. IEEE GLOBECOM Workshops (GC Wkshps), Dec. 2011, pp. 368–372.

[57] H. Wu, et al., “FASA: Accelerated S-ALOHA using access history for event-driven M2M communications,” IEEE/ACM Trans. Network., vol. 21, no. 6, pp. 1904–1917, Dec. 2013.

[58] "Separate backoff scheme for MTC", Proc. 3GPP TSG-RAN2 70bis Meeting, 2010.

[59] MTC simulation results with specific solutions, Madrid, Spain:ZTE, Aug. 2010.

[60] M. S. Ali, E. Hossain, and D. I. Kim, “LTE/LTE-A random access for massive machine-type communications in smart cities,” IEEE Commun. Mag., vol. 55, no. 1, pp. 76–83, Jan. 2017.

[61] 3GPP:R2-103759, “Load distribution for MTC devices,” in Proc. 3GPP TSG-RAN WG2 Meeting, Alcatel-Lucent, Alcatel-Lucent Shanghai Bell, 2010.

[62] C.-H. Wei, R.-G. Cheng, S.-L. Tsao, "Performance analysis of group paging for machine-type communications in LTE networks", IEEE Trans. Veh. Technol., vol. 62, no. 7, pp. 3371-3382, Sep. 2013.

[63] G. Farhadi and A. Ito, “Group-based signaling and access control for cellular machine-to-machine communication,” in Proc. IEEE 78th Veh. Technol. Conf., Sept. 2013, pp. 1–6.

[64] J. S. Kim, S. Lee, and M. Y. Chung, “Efficient random-access scheme for massive connectivity in 3GPP low-cost machine-type communications,” IEEE Trans. Veh. Technol., vol. 66, no. 7, pp. 6280–6290, July 2017.

[65] A. M. AlAdwani, A. Gawanmeh, and S. Nicolas, “A demand-side management traffic shaping and scheduling algorithm,” in Proc. Sixth Asia Modelling Symp., May 2012, pp. 205–210.

[66] R. Ratasuk, N. Mangalvedhe, D. Bhatoolaul, and A. Ghosh, “LTEM evolution towards 5G massive MTC,” in Proc. IEEE Globecom Workshops (GC Wkshps), Dec. 2017, pp. 1–6.

[67] N. K. Pratas, H. Thomsen, C. Stefanovi, and P. Popovski, “Codeexpanded random access for machine-type communications,” in Proc. 2012 IEEE Globecom Workshops, Dec 2012, pp. 1681–1686.

[68] Ferdouse, Lilatul et al. “Congestion and overload control techniques in massive M2M systems: a survey.” Trans. Emerg. Telecommun. Technol. vol. 28, 2017.

[69] "Merits of the slotted access methods for MTC", 2011.

[70] "Comparison on RAN loading control schemes for MTC", 2010.

[71] M. Hasan, E. Hossain, and D. Niyato, “Random access for machineto-machine communication in LTE-advanced networks: Issues and approaches,” IEEE Commun. Mag., vol. 51, no. 6, pp. 86–93, Jun. 2013.

[72] L. M. Bello, P. Mitchell, and D. Grace, “Application of Q-learning for RACH access to support M2M traffic over a cellular network,” in Proc. 20th Eur. Wireless Conf. Eur. Wireless, May 2014, pp. 1–6.

[73] A. H. Mohammed, A. S. Khwaja, A. Anpalagan, and I. Woungang, “Base station selection in M2M communication using Q-learning algorithm in LTE-A networks,” in Proc. IEEE 29th Int. Conf. Adv. Inf. Netw. Appl., Mar. 2015, pp. 17–22.

[74] Y. Ruan, W. Wang, Z. Zhang, and V. K. N. Lau, “Delay-aware massive random access for machine-type communications via hierarchical stochastic learning,” in Proc. IEEE Int. Conf. Commun. (ICC), Paris, France, May 2017, pp. 1–6.

[75] J. Moon and Y. Lim, “Access control of MTC devices using reinforcement learning approach,” in Proc. Int. Conf. Inf. Netw., Jan. 2017, pp. 641–643.

[76] A. Larmo, M. Lindstrom, M. Meyer, G. Pelletier, J. Torsner, and H. Wiemann, "TheLTE link-layer design," IEEE Communications Magazine, vol. 47, pp. 52-59, 2009.

[77] R. Patachaianand and K. Sandrasegaran, "System-Level Modeling and Simulation of Uplink WCDMA," in Fifth International Conference on Information Technology: New Generations (ITNG), pp. 1071-1076, 2008.

[78] H. Harada and R. Prasad, Simulation and Software Radio for Mobile Communications. Boston, London: Artech House, 2002.

[79] T. S. Rappaport, Wireless Communications Principles and Practice, Second ed. Upper Saddle River: Prentice HAll, 2009.

[80] "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and modulation," ed: ETSI, 2009-10.

[81] H. Khedher, F. Valois, and S. Tabbane, "Traffic characterization for mobile networks," in IEEE 56th Vehicular Technology Conference, Proceedings (VTC 2002- Fall.), vol.3, pp. 1485-1489, 2002.

[82] B. M. Kuboye, B. K. Alese, O. Fajuyigbe, and O. S. Adewale, "Development of Models for Managing Network Congestion on Global System for Mobile Communication (GSM) in Nigeria," Wireless Networking and Communications, vol. 1, pp. 8-15, 2011.

[83] B. Chandrasekeran. Survey of network Traffic Models. Available: http:www.cse.wustl.edujaincse56706ftptrafficmodels3index.html accessed April 2012.

[84] A. S. Tanembaum, Computer Networks. Boston, U.S.A.: Pearson Education Inc., 2011.

[85] N. Abramson, "THE ALOHA SYSTEM: another alternative for computer communications," in Proceedings of the, fall joint computer conference, pp. 281-285, November 17-19, 1970.

[86] C. Luders and R. Haferbeck, "The performance of the GSM random access procedure," in 44th IEEE Vehicular Technology Conference, 1994, vol.2 pp. 1165- 1169, 1994.

[87] Q. Li, R. Q. Hu, Y. Qian and G. Wu, "Cooperative communications for wireless networks: techniques and applications in LTE-advanced systems," in IEEE Wireless Communications, vol. 19, no. 2, pp. 22-29, April 2012.

[88] A. Gosavi, “Reinforcement learning: A tutorial survey and recent advances,” INFORMS Journal on Computing, vol. 21, no. 2, pp. 178-192, 2009.

[89] 3rd Generation Partnership Project (3GPP), Evolved Universal Terrestrial Radio Access (EUTRA) and Evolved Universal Terrestrial Radio Access Network (EUTRAN); Overall Description, TS 36.300 v.16.5.0, Mar. 2021.

[90] G. Berardinelli et al., “Reliability analysis of uplink grant-free transmission over shared resources,” IEEE Access, vol. 6, 2018.

[91] A. Azari, P. Popovski, G. Miao, C. Stefanovic, Grant-free radio access for short-packet communications over 5G networks, in GLOBECOM 2017 – 2017 IEEE Global Communications Conference, Dec 2017.

[92] El-Tanab, Manal, and Walaa Hamouda. "An Overview of Uplink Access Techniques in Machine-Type Communications." IEEE Network (2020).

[93] C. Kong, “OPNET Simulation of Signaling System No. 7 (SS7) Network Interfaces”, Storming Media, Page: 256-258, Mar 2000.

[94] I. Katzela “Modeling and Simulating Communication Networks: A Hands-on Approach Using OPNET”, Prentice Hall, Page: 35-38, 1999.

[95] Kernighan B.W, Ritchie D.M, “The C Programming Language, Second Edition”, Prentice Hall, Inc, 1998.

[96] OPNET Documentation Team, "Radio Transceiver Pipeline," Wireless Module User Guide, OPNET Modeler 12.0.A Product Documentation, 2006.

[97] Rampfl, Sebastian. "Network simulation and its limitations." In Proceeding zum seminar future internet (FI), Innovative Internet Technologien und Mobilkommunikation (IITM) und autonomous communication networks (ACN), vol. 57. 2013.

[98] Wael Gorma, “Effective Medium Access Control for Underwater Acoustic Sensor Networks”, PhD thesis, 2019.

[99] Agiwal M, Maheshwari MK, Toor WT. “Two Step Random Access Latency Improvement in Congested beyond 5G Networks”, 2020.