

**RESOURCE ALLOCATION IN CLUSTERED M2M NETWORKS: A  
Q-LEARNING APPROACH**

by

**Fatima Hussain**

B.Sc.Eng., University of Engineering and Technology Lahore, Pakistan, 1998

M.Eng., University of Engineering and Technology Lahore, Pakistan, 2006

MASc., Ryerson University, Toronto, Canada, 2012

A Dissertation

presented to Ryerson University

in partial fulfilment of the requirements for the degree of

Doctor of Philosophy in the

Program of Electrical and Computer Engineering

Toronto, Ontario, Canada

© Fatima Hussain, 2016

## AUTHORS DECLARATION FOR ELECTRONIC SUBMISSION OF A DISSERTATION

I hereby declare that I am the sole author of this dissertation. This is a true copy of the dissertation, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this dissertation to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this dissertation by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my dissertation may be made electronically available to the public.

# Abstract

## Resource Allocation in Clustered M2M Networks: A Q-Learning Approach

© Fatima Hussain, 2016

Doctor of Philosophy

Electrical and Computer Engineering

Ryerson University

Machine to machine (M2M) communication has received increasing attention in recent years. A M2M network exhibits salient features such as large number of machines/devices, low data rates, delay tolerant/sensitive, small sized packets, energy-constrained and low or no mobility. A large number of M2M terminals may exist in a small area with many trying to simultaneously and randomly access for channel resources - which will result in overload and access problem. This increased signaling overhead and diverse requirements of machine type communication devices (MTCs) call for the development of flexible and efficient scheduling and random access techniques. In this thesis, we first review and compare various scheduling and random access techniques in LTE-based cellular networks for M2M communication. We also discuss how successful they are to fulfill the unique requirements of M2M communication and networking. Resource management in M2M networks with a large number devices is also reviewed from the access point of view.

We propose a multi-objective optimization based solution to the problem of resource allocation in interference-limited M2M communication. We consider MTCs in a clustered network structure, where they are divided into clusters and the devices belonging to a cluster communicate to cluster head (or controller). We maximize the number of admitted MTC controllers and throughput with least interference caused to conventional primary users. We formulate the problem as a mixed-integer non-linear problem with multiple objectives and solve it using meshed adaptive direct search (MADS) algorithm. Simulation results show the effects of varying different parameters on cumulative throughput and the number of admitted

MTCD controllers. We then formulate the slot selection problem in M2M networks with admitted MTCDs as an optimization problem.

We present a solution using the Q-learning algorithm to select conflict-free slot assignment in a random access network with MTCD controllers. The performance of the solution is dependent on parameters such as learning rate and reward. We thoroughly analyze the performance of the proposed algorithm considering different parameters related to its operation. We also compare it with simple ALOHA and channel-based scheduled allocation and show that the proposed Q-learning based technique has a higher probability of assigning slots compared to these techniques. We then present a block based Q-learning algorithm for the scheduling of MTCDs in clustered M2M communication networks. At first centralized slot assignment is done and an algorithm is proposed for minimizing the inter-cluster interference. Then we propose to use an Q-learning algorithm to assign slots in a distributed manner and comparison is made between the two schemes. Afterwards, we show the effects of distributed slot-assignment with respect to varying signal-to-interference ratio on convergence rate and convergence probability. Cumulative distribution function is used to study the effect of various SIR threshold levels on the convergence probability. With the increase in SIR threshold levels, increase in convergence time and decrease in convergence probability are observed, as less block configuration fulfills the required threshold in the M2M network.

# Acknowledgment

All praise goes to Allah, who is the most beneficial and merciful.

# Contents

Author's Declaration . . . . .	iii
Abstract . . . . .	v
Acknowledgement . . . . .	vi
<b>1 Introduction</b>	<b>1</b>
1.1 M2M Communication . . . . .	1
1.2 Reinforcement Learning Approach . . . . .	4
1.3 Resource Allocation in M2M Networks . . . . .	5
1.4 Thesis Contribution . . . . .	6
1.5 Thesis Organization . . . . .	6
<b>2 Literature Review: Scheduling and Access in M2M Networks</b>	<b>9</b>
2.1 Introduction . . . . .	10
2.2 Scheduling in M2M Communication . . . . .	14
2.2.1 Channel Awareness in Scheduling . . . . .	15
2.2.2 Delay Dependent Scheduling . . . . .	16
2.2.3 QoS Based Group Scheduling . . . . .	20
2.2.4 Efficient Spectrum Utilization . . . . .	23
2.3 Random Access . . . . .	26
2.3.1 Random Access in MTC Devices . . . . .	28
2.3.2 Access Delay . . . . .	28
2.3.3 Traffic Overload and Congestion Control . . . . .	33

2.4	Resource Management . . . . .	39
2.4.1	Resource Allocation . . . . .	39
2.4.2	Energy-Efficient Algorithms . . . . .	41
2.4.3	Admission and Rate Control . . . . .	48
2.5	Prospective Scheduling and Random Access Techniques . . . . .	50
2.6	Chapter Summary . . . . .	56
<b>3</b>	<b>MTCD Controller Resource Allocation</b>	<b>57</b>
3.1	Introduction . . . . .	57
3.2	System Model . . . . .	59
3.3	Problem Formulation and Solution . . . . .	62
3.3.1	Problem Description . . . . .	62
3.3.2	Proposed Solution . . . . .	66
3.4	Results and Discussion . . . . .	68
3.4.1	Simulation Setup . . . . .	68
3.4.2	Results and Discussion . . . . .	70
3.5	Chapter Summary . . . . .	73
<b>4</b>	<b>Slot Allocation without Interference</b>	<b>77</b>
4.1	Introduction . . . . .	78
4.2	System Model . . . . .	81
4.2.1	Clustered M2M Network . . . . .	81
4.2.2	Slot/Channel Allocation . . . . .	83
4.2.3	Formation of Clusters . . . . .	84
4.3	Problem Formulation . . . . .	85
4.3.1	Single Frequency Case . . . . .	86
4.3.2	Multiple Frequency Case . . . . .	89
4.4	Proposed Solution . . . . .	90
4.4.1	Classic Techniques . . . . .	90

4.4.2	Reinforcement learning . . . . .	91
4.4.3	Q-Learning Algorithm for M2M Communication . . . . .	91
4.5	Results and Discussions . . . . .	94
4.5.1	Effect of Learning Rate $\alpha$ . . . . .	95
4.5.2	Effect of Reward $R$ . . . . .	96
4.5.3	Effect of Both $\alpha$ and $R$ . . . . .	99
4.5.4	Q-Learning versus RA/CBA . . . . .	102
4.6	Chapter Summary . . . . .	103
<b>5</b>	<b>Slot Allocation with Interference</b>	<b>105</b>
5.1	Introduction . . . . .	106
5.2	System Model . . . . .	110
5.3	Proposed Solution . . . . .	112
5.3.1	Centralized Slot Allocation . . . . .	113
5.3.2	Distributed Slot Allocation . . . . .	117
5.4	Results and Discussion . . . . .	125
5.4.1	Centralized vs Distributed Implementation . . . . .	125
5.4.2	Distributed Implementation . . . . .	126
5.5	Chapter Summary . . . . .	129
<b>6</b>	<b>Conclusions and Future work</b>	<b>138</b>
6.1	Conclusion . . . . .	138
6.2	Future Work . . . . .	139

# List of Tables

3.1	Notations . . . . .	61
3.2	Common Parameter Values . . . . .	69
4.1	Notations . . . . .	82
4.2	K-Means Algorithm . . . . .	85
5.1	Algorithm for SIR Based Slot Assignment . . . . .	114
5.2	Colour Assignment . . . . .	116
5.3	Comparison of Reward Schemes . . . . .	131
5.4	Comparison between Centralized and Distributed Schemes . . . . .	132
5.5	Notations and Values . . . . .	133

# List of Figures

1.1	Thesis Organisation . . . . .	8
2.1	M2M with Diverse Network Applications . . . . .	11
2.2	Scheduling Objective Types. . . . .	17
2.3	Random Access Steps [1] . . . . .	27
2.4	Random Access Objective Types . . . . .	32
2.5	Virtual Resource Allocation For MTC Classes [2] . . . . .	36
2.6	Coordinator Random Access [3]. . . . .	44
3.1	Heterogenous Network with eNB, MTCDS and Controllers . . . . .	60
3.2	Layout for M=2 . . . . .	70
3.3	Sum Rate and Admitted MTCD for different set of weights: Number of HDs = 1, With the increase in the $w_1$ , the sum rate will increase. (a) $[w_1 w_2 w_3] = [0.1 0.3 0.6]$ , (b) $[w_1 w_2 w_3] = [0.33 0.33 0.3]$ , (c) $[w_1 w_2 w_3] = [0.5 0.25 0.25]$ , (d) $[w_1 w_2 w_3] = [0.8 0.1 0.1]$ . . . . .	74
3.4	Sum Power and Sum Rate for different set of weights : Number of HDs = 1, Less value of $W_2$ means less importance to power optimization. (a) $[w_1 w_2 w_3] = [0.1 0.3 0.6]$ , (b) $[w_1 w_2 w_3] = [0.33 0.33 0.3]$ , (c) $[w_1 w_2 w_3] = [0.5 0.25 0.25]$ , (d) $[w_1 w_2 w_3] = [0.8 0.1 0.1]$ . . . . .	75
3.5	Admitted MTCD for different rate and interference. (a) $[w_1 w_2 w_3] = [0.8 0.1 0.1]$ , Interference = 10mW (b) $[w_1 w_2 w_3] = [0.8 0.1 0.1]$ , Interference = 100mW: Less interference threshold means less number of admitted MTCDS . . . . .	76

3.6	Admitted MTC D for different number of human users: Less HDs means more number of admitted MTC Ds $[w_1 \ w_2 \ w_3] = [0.33 \ 0.33 \ 0.33]$ , Interference = 10mW . . . . .	76
4.1	Frame Example . . . . .	87
4.2	Collision in Slots . . . . .	87
4.3	Slot Search Flow Chart . . . . .	94
4.4	Clustered M2M Network . . . . .	95
4.5	Variable Learning Rate; dots show the actual data points and line shows the curve fitted data . . . . .	97
4.6	Convergence Probability . . . . .	98
4.7	Variation in Both Negative and Positive Rewards . . . . .	99
4.8	Variable Positive Reward . . . . .	100
4.9	Variable Negative Reward . . . . .	100
4.10	Variation in Positive Reward with Different Negative Rewards at $\alpha = 0.1, 0.9$ . dotted line: $\alpha = 0.1$ , solid line: $\alpha = 0.9$ . . . . .	101
4.11	Variation in Negative Reward with Different Positive Rewards at $\alpha = 0.1, 0.9$ . dotted line: $\alpha = 0.1$ , solid line: $\alpha = 0.9$ . . . . .	101
4.12	Comparison between Q-Learning and Channel Based Allocation . . . . .	102
5.1	Clustered M2M Network . . . . .	110
5.2	Frame Allocation and Interference. ( $T = 3, K = 3, D = 3$ ) . . . . .	111
5.3	Interference Graph . . . . .	114
5.4	Slot Assignment . . . . .	117
5.5	Block Q-learning Flow Chart . . . . .	130
5.6	Evolution of Q-values vs Iteration for Various Rewards . . . . .	132
5.7	Admittance Rate for Centralized and Distributed Allocation . . . . .	133
5.8	Average SIR for Centralized and Distributed Allocation . . . . .	134

5.9	Convergence Probability at Various Reward Schemes; R1:Pessimistic, R2: Optimistic, R3:Balanced . . . . .	134
5.10	Convergence Time at Various Reward Schemes; R1:Pessimistic, R2: Optimistic, R3:Balanced . . . . .	135
5.11	Convergence Time at Various SIR Thresholds . . . . .	135
5.12	Convergence Probability at Various SIR Thresholds . . . . .	136
5.13	Successful MTCDs at Various SIR Thresholds . . . . .	136
5.14	Successful MTCDs at SIR Threshold 2, 6, 8 . . . . .	137

# List of Abbreviations

AGTI	Access Grant Interval
AP	Access Point
ACB	Access Class Barring
ACK/NAK	Positive/Negative Acknowledgment
BER	M2M outnumbered human end users
BER	Bit Error Rate
CQI	Channel Quality Indicator
CSI	Channel State Indicator
CDMA	Code Division Multiple Access
DRX	Discontinuous Reception
EAB	Extended Access Barring
eNB	Evolved Node B
4G	Fourth Generation
GPRS	General Packet Radio Service
GPS	Global Positioning System
GSM	Global System for Mobile Communication
GTI	Granted Time Interval
H2H	Human to Human
ISI	Inter symbol Interference
M2M	Machine to Machine
MTC	Machine Type Communication
MAC	Medium Access Control
MACA	Multiple Access with Collision Avoidance
MAI	Multiple Access Interference
MUE	Machine User Equipments

MPR	Minimum Packet Rate
OPR	Offered Packet Rate
OFCDM	Orthogonal Frequency and Code Division Multiplexing
OFDM	Orthogonal Frequency Division Multiplexing
PHY	Physical Layer
PRB	Physical Resource Blocks
PUCCH	Physical
PF	application specific Yes
PRACH	Physical Random Access Channel
QoS	Quality of Service
RA	Random Access
RRM	Radio Resource Management
RNTI	Radio Network Temporary Identifier
RACH	Random Access Channel
SIR	Signal to Interference Ratio
SINR	Signal to Interference and Noise Ratio
SNR	Signal to Noise Ratio
SIC	Successive Interference Cancellation
RFSA	Reservation Frame Slotted Aloha
TDMA	Time Division Multiple Access
TTI	Transmission Time Interval
TA	Timing Alignment
UMTS	Universal Mobile Telecommunication System
UL	Uplink
WSN	Wireless Sensor Network
WiFi	Wireless Fidelity
WiMAX	Wireless Inter-operability for Microwave Access

# Chapter 1

## Introduction

Mobile communication systems have evolved from supporting analog voice only to rich multimedia services providing hundreds of thousands of different applications to billions of users. We are just at the beginning of a transition into a fully connected networked society that will provide access to information and sharing of data anywhere and anytime for anyone and anything. Thus, in the future wireless access will not only be about connectivity for people but for anything that benefits from being connected. This includes such diverse things as household appliances, traffic control and safety functions, infrastructure monitoring systems, medical equipment, and much more. As a consequence, compared to the wireless networks of today, next-generation wireless access will support a much wider range of use case characteristics and corresponding access requirements [4].

### 1.1 M2M Communication

Transition to a networked society will lead to a massive number of connected devices, and cellular network-based machine-to-machine (M2M) communications is fast becoming a market-changing force for a wide spectrum of businesses and applications such as telematics, smart metering, point-of-sale terminals, and home security and automation systems. M2M refers to set of technologies that are used to connect systems for the purpose of remote monitor-

ing and control without human intervention. To connect end devices with the surrounding infrastructure, M2M communications typically rely on cellular or low-power wireless technologies [5].

Compared to traditional automation technologies, one major difference for this new generation of smart devices is how tightly they are coupled into larger-scale service infrastructures [6]. For example, in logistic operations, the locations of fleet vehicles can be tracked with automatic vehicle location devices and uploaded into back-end automatic dispatching and planning systems for real-time global fleet management. More and more emerging technologies also heavily depend on these smart devices. For instance, with smart grid, individual customer's power usage on a real-time and wide-area basis can be controlled through devices such as the electric meters modules.

It is estimated that there are already tens of millions of such smart devices connected to cellular networks worldwide, and within the next five years, this number will grow to hundreds of millions. Various countries are embracing the M2M technology for different applications. For example, the European Union's goal is to have every vehicle equipped with the eCall system by 2015. In Brazil, all new vehicles produced in, or imported to, will be equipped with GPRS modules to reduce vehicle theft. The UK government vision is for every home to have smart meters by 2020 [7]. Another reason for M2M growth is that it offers something to each entity in the ecosystem from chip set providers to M2M module makers to network operators to system integrators. To the enterprises, it offers a common technology infrastructure for providing value-added service offerings such as efficient managed services and faster technical resolutions.

Currently, traditional cellular networks have been considered as one of the choices to bear M2M communication [8,9]. It has an advantage of large coverage area and economical for operators in a single network. However, current cellular networks are designed for human to human (H2H) communication which aim at voice and media transmission, with real time, small delay, and high throughput. M2M communication over cellular networks poses significant challenges as a result of large number of devices, small data transmissions, and

vast applications range. Potential issues are on the air interface, including physical layer transmissions, the random access (RA) procedure, scheduling and radio resource allocation with massive number of machine type communication devices (MTCs).

One of the important and challenging issues in M2M deployment, is to reduce the access delay in simultaneous channel access attempts from massive number of MTCs. Both machine devices and mobile users may suffer from severe collisions during network entry/reentry, handover, scheduling request, and uplink timing synchronization. As a result, they have to backoff their transmissions. This backoff and resulting retry attempts further add to the access delay.

Important task is to efficiently handle MTC traffic characteristics, specifically the load generated by massive simultaneous low data rate transmissions. Therefore, network should have the capability of counteracting signaling and traffic load spikes caused by a sudden surge of massive numbers of MTCs trying to access the eNB all at once. This massive access can be due to several devices requesting to transmit at the same time by some applications or due to roaming MTC devices suddenly attaching due to a fault in their home network thus causing overload. In such situations, radio and signalling network congestions may occur due to massive concurrent transmissions which can lead to large delays, packet loss and, in the extreme case, service unavailability. M2M communications have different characteristics from traditional Internet traffic, and applying traditional congestion control techniques to M2M traffic may yield suboptimal results. Various overload control techniques are defined by 3GPP such as M2M server triggering eNB for paging M2M terminals, broadcast-based, rejecting connection requests in the radio access network and in the core network etc.

Efficient, fair and less complex resource allocation is important in M2M networks with distributed implementation for a large number of MTCs. Fixed channel allocation though easy to implement often cannot satisfy the unbalanced and dynamic traffic demands in M2M networks [10]. It has to be distributed and dynamic in nature. Also, resource allocation plays an important role in interference management among MTCs and human users.

## 1.2 Reinforcement Learning Approach

Reinforcement learning is a trial-and-error approach to learning in which an agent operating in an environment learns how to achieve a task in that environment. Original reinforcement learning came out of the machine learning community, but where machine learning normally is a type of supervised learning, reinforcement learning is not. In supervised learning methods, whenever an action is taken, the learner tells the agent afterwards what the best action would have been. The agent can then adjust its behaviour according to this supervision. In contrast to this, the reinforcement learning agent is just given a scalar reinforcement signal as a response to its action, but no clue on what the best action would have been. It has to find a good strategy for increasing the sum of the reinforcement signals by trial-and-error interactions with the environment. This important distinction indicates that reinforcement learning is well-suited for problems where the correct behaviour of the agent is not known in advance, but can be derived from the trial-and-error interactions with a perhaps dynamic environment.

In short, reinforcement learning enables an agent (e.g., a sensor node in WSN network or smart device in M2M network) to learn by interacting with its environment. The agent will learn to take the best actions that maximize its long-term rewards by using its own experience. The most well-known reinforcement learning technique is Q-learning [11]. In Q-learning, an agent regularly updates its achieved rewards based on the taken action at a given state. The future total reward (i.e., the Q-value) of performing an action  $a_t$  at a given state  $s_t$  is computed:

$$Q(s_{t+1}, a_{t+1}) = Q(s_t, a_t) + \alpha(r(s_t, a_t) - Q(s_t, a_t)), \quad (1.1)$$

where  $r(s_t, a_t)$  denotes the immediate reward of performing an action  $a_t$  and  $\alpha$  is the learning rate that determines how fast learning occurs (usually set to value between 0 and 1).

Machine learning techniques are widely applied to protocol design due to their ability to deal with very complex systems in a relatively simple and efficient manner. The management

of M2M massive access is no exception, and several studies in this domain have adopted machine learning techniques to address different problems.

### 1.3 Resource Allocation in M2M Networks

Due to the diversity of M2M applications, M2M traffic shows different characteristics and random patterns, which differ M2M traffic from conventional cellular traffic. It is comprised of time-controlled and delay-tolerant traffic, synchronized and periodic traffic, and infrequent, bursty and one way data traffic. In general, the M2M devices generate more uplink traffic than downlink in contrast to the existing cellular system.

Resource allocation and management is a challenging task in M2M networks due to the diversity of M2M applications, special features and requirements of MTCs specifically become more critical when communication is done through cellular networks. Resource management, sharing, scheduling and prioritization techniques need to be addressed properly for the efficient M2M communication and with least interference caused to conventional traffic. Therefore, attention must be focused on the design of resource allocation scheme, and effort should be taken to make eNB more intelligent.

Implementing and optimizing the wireless resource sharing between the M2M and H2H links is an challenging task. An orthogonal resource allocation is simple to implement and does not cause any interference among both types of users. However, the resource efficiency increases when using non-orthogonal sharing. Where both types of users are sharing the same frequency and time, it results in interference thus costing efficiency.

The authors in [12], propose energy efficient radio resource allocation method for MTC-enabled LTE networks in which they model resource allocation problem as sum power minimization problem. Similarly, the authors in [13] propose a max-utility scheduling method for machine-type communication by considering QoS requirements as minimizing co-channel interference between H2H and MTC devices. Group based allocation [3] and joint access control and resource allocation method [14] consider fairness and random access efficiency

in their allocation procedure. In this scheme, PRACH resources such as preamble sequences and RA slots are separately allocated to M2M terminals and H2H terminals [15]. If the the M2M terminals share the same PRACH resources as H2H terminals, then PRACH overload condition could significantly affect RA quality of H2H terminals.

## 1.4 Thesis Contribution

In this thesis, we utilize reinforcement learning for slot allocation in MTCDs for maximizing throughput with minimum interference in clustered M2M networks. The main contributions of the dissertation are:

- Provided a comprehensive review of channel access and resource allocation techniques for emerging heterogenous networks with M2M and H2H traffic, that covers both scheduling and random access methods for MTCDs from various metrics perspectives.
- Formulated the problem of resource allocation in interference-limited M2M communication as a mixed-integer non-linear problem having multiple objectives, and solved it using mesh adaptive direct search algorithm.
- Proposed a distributed self-organized conflict-free slot allocation scheme based on the paradigm of independent learning, where agents are unaware of the other agents' actions in a clustered M2M network.
- Proposed and analyzed a centralised and distributed block-based Q-learning algorithm for slot assignment in MTCDs that minimizes inter-cluster interference.

## 1.5 Thesis Organization

Chapter 2 summarizes various scheduling techniques and protocols considered in literature for MTCD with critical analysis. Also, various random access methods for MTCDs contending for channel access are discussed with a critical requirement of shared medium supporting

higher load density. Resource allocation and energy efficiency in massive M2M communications are also discussed.

In Chapter 3, we consider a clustered network consisting of primary H2H users (PUs) and MTCs, where all MTCs communicate with a cluster head. We propose a multi-objective optimization based solution to the problem of resource allocation for maximizing the number of admitted MTC controllers and throughput with least interference caused to conventional H2H users.

In Chapter 4, reinforcement learning is introduced and is proposed for slot allocation of MTCs. Convergence capabilities such as convergence time, convergence probability and cumulative success rate are studied and simulation results of the proposed approach with respect to different parameters are presented.

In Chapter 5, a block-based Q-learning algorithm is introduced for scheduling of MTCs in clustered M2M communication networks. It is used to assign slots to devices in a distributed manner. In addition to this, an centralized algorithm for the slot allocation of MTCs is proposed and comparison is made between two methods.

Finally, 6 concludes the thesis with summary of the work and potential future work.

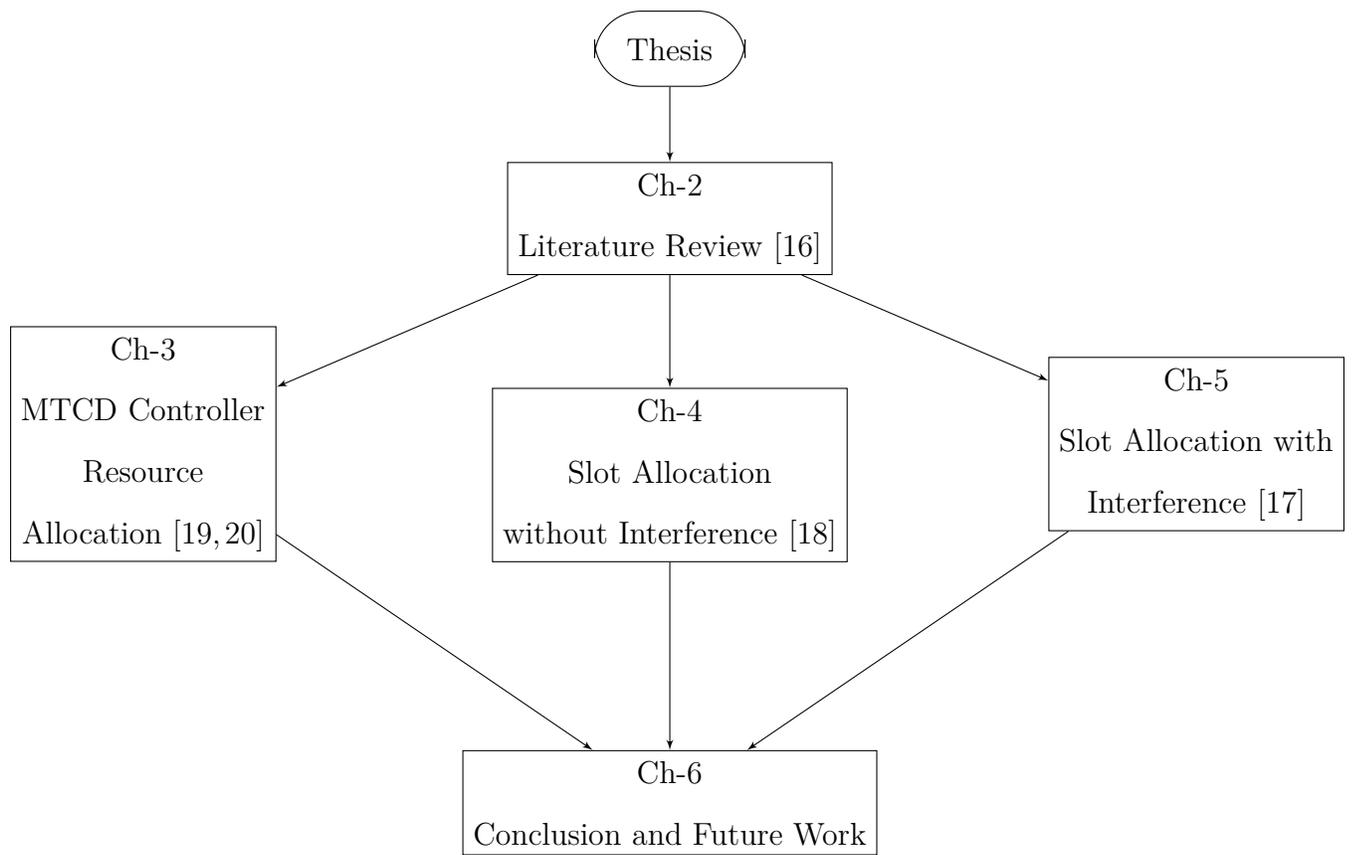


Figure 1.1: Thesis Organisation

# Chapter 2

## Literature Review: Scheduling and Access in M2M Networks

Machine to machine (M2M) communication has received increasing attention in recent years. A M2M network exhibits some salient features such as large number of machines/devices, low data rates, delay tolerant/sensitive, small sized packets, energy-constrained and low or no mobility. A large number of M2M terminals may exist in a small area with many trying to simultaneously and randomly access for channel resources - which will result in overload and access problem. This increased signaling overhead and diverse requirements of machine type communication (MTC) devices call for the development of flexible and efficient scheduling and random access techniques. In a M2M scenario, where the network is operating at high offered load with a large number of contending transmitters, distributed random access techniques are more appropriate than centralized scheduling techniques due to less control messages and better channel utilization. There is a need for comparison of various medium access methods that can be used in the development of an efficient hybrid M2M and human to human (H2H) network.

In this chapter, we review and compare various scheduling and random access techniques in cellular networks, particularly in LTE based networks. We also discuss how successful they are to fulfill the unique requirements of M2M communication and networking. Resource

management in M2M networks with a large number of MTC devices is also discussed from the access point of view. Energy efficiency, being one of the main challenges of QoS constrained M2M communication, is also discussed. Minimization of the energy consumption is tightly bound to channel access and hence considered in the comparison of various MAC protocols. Finally, some potential research directions related to access control and resource allocation are presented for future work.

## 2.1 Introduction

Machine to machine communication is defined in different ways and contexts in the literature. A broader definition of M2M communication includes the remote control of machines (telematics), monitoring and collecting data from machines (telemetry). Recently, from a mobile perspective, M2M is defined as communication between a machine and a mobile terminal, or between a machine and a back-end information system [21]. The broader discipline of M2M communications is experiencing rapid growth across the globe. Due to the advancement in inexpensive sensors and devices, and mobile communication technologies, various M2M applications are emerging. This is noticed in a number of broad domains such as, energy management, transportation infrastructure [22], smart city [23–25], smart home [26, 27], vehicular telematics [28, 29], medical/health services [30–32] and information management [33–35], and industrial environment [36] as shown in Figure 3.1.

However, there are still significant barriers for its growth such as security, privacy, service capabilities, testing, and certification of devices that need to be overcome before the M2M market can reach its full potential [37]. Therefore, a number of standardization bodies and regulatory authorities are striving hard for end-to-end connectivity and streamlining the fragmented M2M market. European telecommunication standard institute (ETSI) [38, 39] aims at developing and maintaining an end-to-end architecture for M2M systems. It also addresses various issues of naming, addressing, QoS, security, charging, application interfaces and hardware interfaces [40, 41]. Also, 3rd generation partnership project (3GPP) technical

specifications group dealing with service and system aspects, has issued a number of specifications dealing with integration and support of M2M communications [42]. The international telecommunication union (ITU-T) addresses the area of networked intelligent sensors, covers the issue of M2M communication with interconnected sensor to other networks.

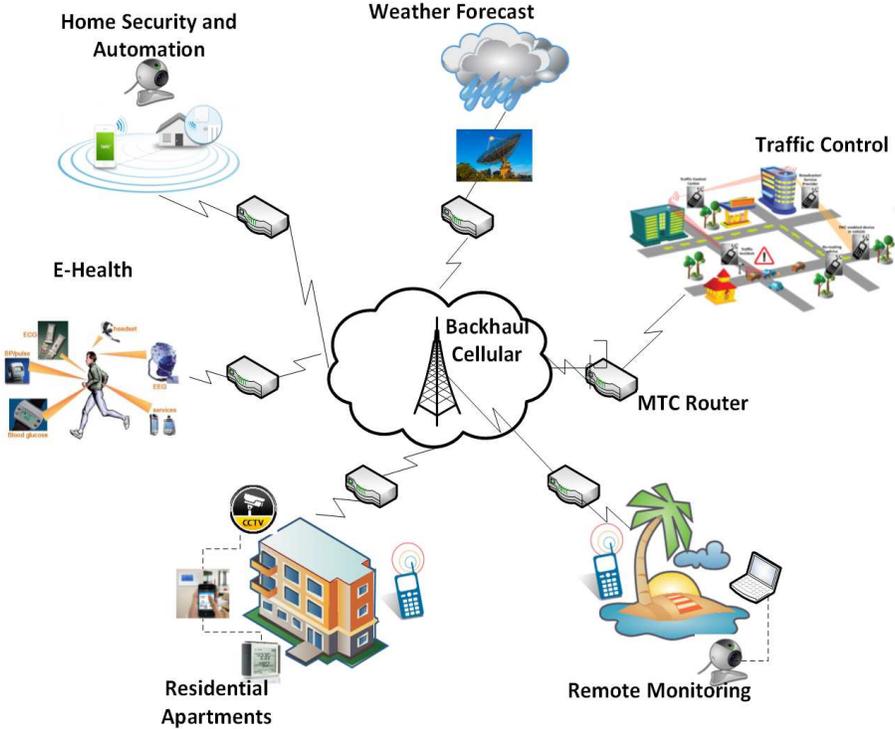


Figure 2.1: M2M with Diverse Network Applications

MTC devices are typically small, inexpensive, able to operate unattended by human for extended period of time, and communicate over the wireless area network. The M2M network includes a large number of diverse often interconnected smart devices with the goal to provide reliable connections among them. However, in M2M communication, there can be a large number of MTC devices, each with only a small amount of data needing to be transmitted. Therefore, air interface design already available for high data rate transmissions does not effectively support this type of communication.

Currently, traditional cellular networks have been considered as one of the choices to bear M2M communication [8,9]. It has an advantage of large coverage area and economical for

operators in a single network. However, current cellular networks are designed for human to human (H2H) communication which aim at voice and media transmission, with real time, small delay, and high throughput. M2M communication over cellular networks poses significant challenges as a result of large number of devices, small data transmissions, and vast applications range. Potential issues are on the air interface, including physical layer transmissions, the random access (RA) procedure, scheduling and radio resource allocation with massive number of MTC devices.

Resource allocation and management is expected to play an important role in the deployment of MTC devices. Medium access control (MAC) layer allocates radio resources to users according to several parameters such as source traffic characteristics, required QoS, frequency, time and space diversity of the used channel [43]. MAC plays a key role in determining the throughput, delay, and power consumption of wireless networks. A MAC design that is adaptable to the changing traffic and channel characteristics is required. Therefore, the system performance can be improved by exploiting the radio resources more efficiently.

Before we go into details of various RA and scheduling techniques, we will briefly discuss various technologies used for M2M communication so far. If channelization for M2M MAC is considered, contention-based carrier sense multiple access (CSMA) and schedule-based TDMA schemes are the traditional choices. CSMA due to its simplicity and flexibility, is the natural choice. But its performance can be significantly degraded in situations of high contention due to the high overhead accrued for resolving collisions. In contrast, TDMA can generally achieve higher network throughput in situations of high contention as well as energy efficiency [44]. However, scheduling the time slots is very difficult to achieve in a distributed fashion for uplink M2M communication due to the stricter synchronization requirement between each device and M2M base station. This thesis mainly address the problem of distributed slot allocation in M2M networks.

A large number of MAC protocols previously used in wireless sensor networks (WSNs) are proposed in one form or the another for MTC [45–48]. Although the use of WSNs in M2M communication can bring significant advantages, M2M communication properties impose

unique challenges including reliability, latency, packet errors and variable link capacity, under harsh environmental condition, and resource constraints [49].

GPRS capabilities can be used to support some M2M applications since GPRS provides a manageable and cost-effective way for M2M deployment [50, 51]. GPRS could also be considered for MTC as it can support bursty traffic such as generated by several M2M applications. However, it has several limitations, which raises serious considerations for its suitability for future M2M applications. As long as the number of MTC devices is small, GPRS is an adequate solution. But as the number increases, it may be inappropriate. Because, GPRS capacity is limited for supporting the emerging M2M applications and services with thousands of devices per cell [52]. Since the end devices must compete for the channel, it suffers from degraded performance in terms of throughput and energy consumption, and due to the high probability of collision at higher load [53].

Evolving LTE-A is expected to play a central role in interconnecting machines, and is recommended to efficiently cater to M2M communications [52]. It offers higher capacity and more flexible radio resource management (RRM) compared to GPRS. M2M is characterized by a high device density in a cell, small amounts of payload, machine originated communications and low traffic volumes per machine. LTE for M2M communication can be a conceivable choice due to its longevity, lower service cost and scalability [54]. However, LTE has been originally designed for broadband applications such as voice calls, video streaming, online gaming, social networking and web surfing; while most M2M applications transmit and receive small amounts of data, leading to an unreasonable ratio between payload and required control information. Besides the need for supporting a large number of M2M devices, the research efforts focus on addressing the vast diversity of M2M service characteristics, need for low-energy devices or low latencies and coexistence with current communication systems [3].

The main focus of this chapter is to present and analyze channel access techniques for emerging M2M networks and resource allocation, from medium access point of view, in cellular networks in particular LTE. It covers both scheduling and random access methods

for massive access of MTC devices from various metrics perspectives. The choice of MAC has a strong influence on the energy efficiency of the network, and thus the lifetime of the devices. Therefore, energy efficiency analysis for various protocols is also done. We also identify open challenges in various domains of machine type communications and discuss prospective research. Comparison among various scheduling and random access techniques is done and corresponding pros and cons are discussed.

This chapter is organized as follows. Section II summarizes various scheduling techniques and protocols considered in literature for MTC access with critical analysis. In Section III, various random access methods for MTC contending for channel access are discussed with a critical requirement of shared medium supporting higher load density. Section IV discusses resource allocation and energy efficiency in massive M2M communications. Section V covers some prospective scheduling and random techniques in M2M networks. Section VI concludes the chapter.

## **2.2 Scheduling in M2M Communication**

Scheduling techniques are employed to achieve high spectral reuse and diversity gain in wireless systems in addition to provide QoS guarantees to the users. Several scheduling algorithms proposed for M2M have already been in use in other applications. The emerging scheduling techniques should efficiently accommodate both traditional H2H traffic and the MTC device traffic with different QoS requirements in a hybrid M2M/H2H networks for cost effective operations.

Channel access involves establishment of radio bearers (RB) by first using a common random access channel (RACH) to send relevant control information such as the identity of an device followed by higher layer signalling messages and application data over scheduled dedicated resources. Packet schedulers for both the downlink and the uplink are deployed at the evolved node B (eNB), and it is in charge of performing both radio resource management (RRM) and control procedures on the radio interface. Users are informed about the resource

allocation decisions on a subframe basis through control channels [55]. Data channel (i.e., the PDSCH) is shared among the users, meaning that portions of the spectrum should be distributed at every transmission time interval (TTI) among them.

Various scheduling techniques differ in terms of input connection parameters, objectives, and service targets. In machine-type communication, researchers have classified the scheduling mechanism on the basis of channel awareness, data rate, target delay, QoS provisioning etc. Figure 2.2 summarizes various scheduling objectives considered in the literature for M2M paradigm.

### 2.2.1 Channel Awareness in Scheduling

Channel awareness is a fundamental concept for achieving high performance in a wireless environment. It can be exploited in radio resource management using channel quality indicator (CQI). Since the scheduling changes spatially and timely with each time slot, information from neighboring eNBs is required for data decoding. This can be accomplished by the exchange of channel state information (CSI) and other control signals among all eNBs. For a cellular network consisting of multiple eNBs, however, such cooperation of instantaneous CSI is problematic. It requires precise synchronization among eNBs, and estimation of CSI for multiple cells create immense delays and large communication overhead [56].

Channel aware scheduler can estimate the channel quality perceived by each user and in turn, can predict the achievable throughput. If one can estimate the channel quality perceived by a user on a given resource block, it is possible to allocate radio resources to achieve high data rate. Maximum throughput scheduler aims to assign all those RBs to users that maximize their throughput. It results in unfair resource sharing for users with poor channel conditions. To provide fairness, proportional fair (PF) scheduler is used which maximizes fairness at the expense of the throughput. In contrast to traditional scheduling algorithms such as PF, which overlooks the users's heterogeneous demands, incorporating a proactive and traffic-aware scheduler exhibits gains [57]. Several other channel aware

schedulers such as joint time and frequency domain scheduler, delay sensitive schedulers, buffer-aware schedulers etc. are used for resource scheduling in LTE networks.

One of the main challenges of M2M networks is energy efficiency. We discuss this channel aware techniques in that perspective. System information such as CSI and occupancy level of the queues (QSI) are very helpful in the design of the scheduling policy. Performance goals such as ensuring required QoS, providing fairness, maximizing throughput and minimizing energy consumption can be achieved with specific scheduler design. Main focus of research in the design of M2M schedulers so far is maximizing throughput, reduce delay and provide fairness. The designs based on the above are not energy efficient as effort is not made for minimizing the overall energy consumption in M2M data transfer. However, based on current CSI of each device, future energy usage can be estimated and that can be used for efficiently scheduling heterogeneous traffic with minimum transmission energy.

### **2.2.2 Delay Dependent Scheduling**

To accommodate the delay limited M2M traffic, latency constrained MAC scheduler is crucial in providing necessary physical resources to ensure their proper operation. Delay is a significant QoS parameter and it has a notable impact on the number of users that can be effectively served [58]. Capacity of the system can be improved by including the end to end delay constraint in network design. There is little work done for latency constraint M2M applications mainly based on packet delay budget of MTC devices considering channel conditions.

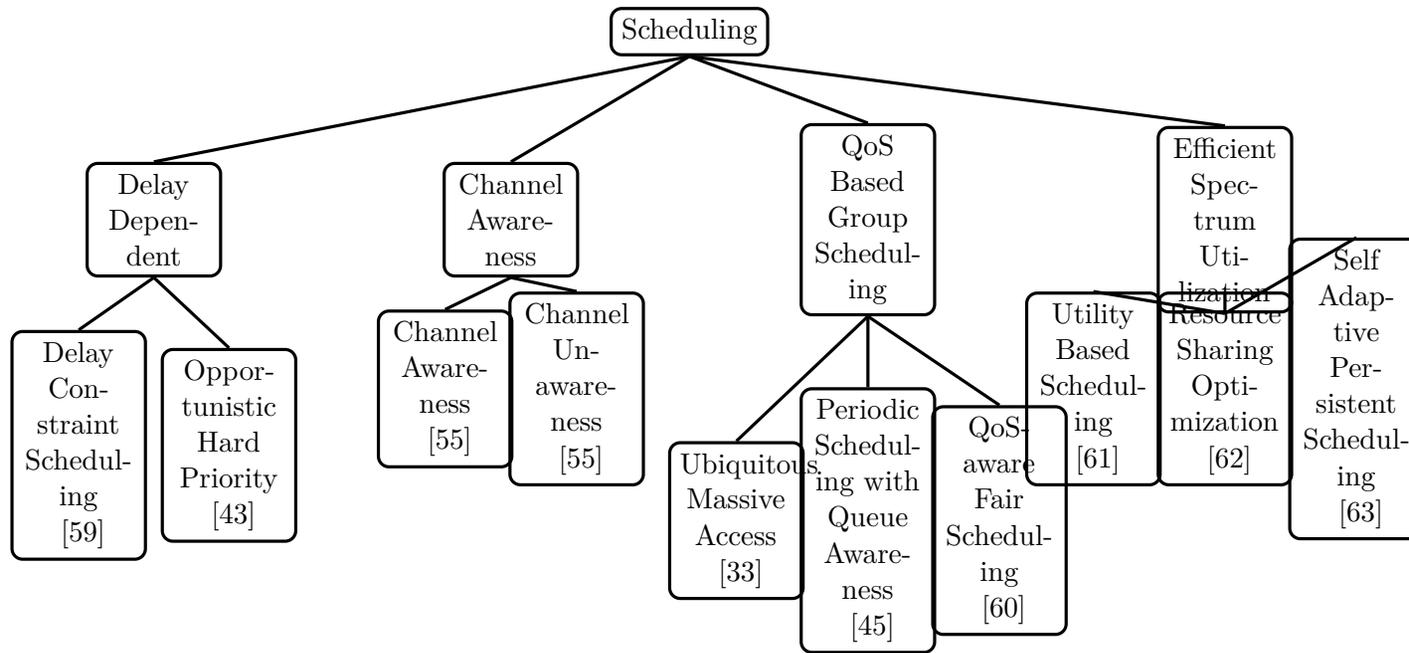


Figure 2.2: Scheduling Objective Types.

## Opportunistic Hard Priority

In [43], authors proposed a resource allocation algorithm keeping in view of the delay constraint M2M applications. In this algorithm, delay threshold and delay budget are assigned to each delay sensitive traffic flow. Same priority is set to all the packets as long as the packet delay is below a certain delay threshold. Packets exceeding the threshold are set higher priority until they are served. The delay threshold is kept lower than the delay budget so that there is a sufficient margin for the scheduler to serve the packets without violating the link budget. Arrival of each packet is in cyclic order and their time in queue is monitored. If it is greater than the delay threshold, it will be served with the highest priority.

This scheduler works well for the higher values of signal-to-noise ratio (SNR) as there are less number of packets whose waiting time exceeds the delay threshold. Therefore, all the packets are set to same priority. But at very lower values of SNR, most of the packets are discarded due to the increase in delay compared to the delay threshold. In such conditions, robust coding schemes are required to achieve target bit error rate.

Authors in [43] proposed another algorithm in which priority to the packets is assigned using two components. One is delay aware earliest deadline due (EDD) and another is channel aware PF. In EDD, users are prioritized according to packet delay level closer to delay budget, while PF term will favor terminals with good channel conditions. The scheduling tags are assigned to the users on the basis of their average throughput, waiting time and maximum allowable delay. If they experience delay, EDD term dominates; otherwise, PF term dominates the scheduling tag. Packets that exceed the delay budget are discarded. The objective is to ensure that instantaneous packet delay is kept below a certain value.

At higher SNR values, this scheduler works well but as the SNR drops to a lower value, throughput per user is very low due to higher outage probability. Also, discarding those packets that exceed the delay budget can result in QoS deterioration. Instead, increment in the delay of the remaining packets to be served, can be done.

Enhancement to this algorithm is made by delaying the EDD term. It improves the

results over the entire range of SNR. It will work as a PF algorithm as long as packet delays are far from exceeding the delay budget. This is the situation of high SNR value. When the SNR is very low, delay is increased due to increased error probability. Therefore, packet delay is above the delay budget and EDD term will give higher priority to the user, thus decreasing the average packet delay. This scheme takes advantage of delay tolerance as a result increasing the system capacity.

### Delay Constrained Scheduling

Two uplink scheduling algorithms considering channel conditions and maximum allowable delay of each MTC device have been proposed in [59, 64]. These are dynamic QoS and channel-aware heuristic scheduling algorithms optimized on a per-TTI basis. First algorithm assigns more weight to channel quality of the user while the second to maximum delay tolerance. Authors assume that  $M$  MTC devices are accessing the shared medium along with  $L$  LTE users in the given TTI, through  $F_k (k = 1, 2, \dots, M)$  and  $T_i (i = 1, 2, \dots, L)$  resource blocks (RBs), respectively. Moreover, maximum delay tolerance of the  $k^{th}$  device is  $D_k$  and SNR seen by it over  $l^{th}$  resource block  $RB_l$  is  $\gamma_{(k,l)}$ .

- In Algorithm 1, maximum delay tolerance of each device along with the conventional scheduler design is considered. Priority is given to existing users to access the total RBs and the rest is available to MTC devices. MTC devices are organized in descending order with respect to  $\gamma_{(k,l)}$ . Allocation of RBs with best SNR  $\gamma_{(k,l)}$  i.e.,  $RB_l$  to the  $k^{th}$  MTC device is done only if maximum delay tolerance is smaller than the mean delay tolerance of all the devices requesting access.  $F_k$  is reduced by one and this process is repeated till  $F_k = 0$  and  $\gamma_{(k,l)}$  is set to 0. Thus, corresponding device cannot compete for another RB but can compete for in adjacent RBs.
- In Algorithm 2, priority is given to devices with low delay tolerance. Best RBs in terms of channel quality are found and assigned to them. Priority is also given to existing users to access the total RBs and rest is available to MTC devices. MTC devices are

organized in ascending order with respect to  $D_k$ . For the MTC of lowest  $D_k$ , RB with the best SNR  $\gamma_{(k,l)}$  is found and is assigned to the device. This is done only if this is adjacent in frequency with the previous RB and decrement  $F_k$  by one. This process is repeated till  $F_k = 0$  and  $\gamma_{(k,l)}$  is set to 0, that means corresponding MTC device cannot compete for another resource block.

The above work has taken MTC in more realistic way by considering the delay constraint of each device, thus increasing the number of served MTC devices. Also, by exploiting the channel quality of each device, cell throughput increases through multi-user diversity. Both algorithms can serve the massive number of MTC devices as compared to conventional channel aware scheduler but at the expense of decrease in bit error rate (BER) and increased signalling overhead. This works well for the low rate applications but will fall behind in case of high rate applications.

### 2.2.3 QoS Based Group Scheduling

To support a large number of MTC devices with small data transmissions and diverse QoS requirements is a difficult task. MAC scheduler is an important and crucial entity responsible for efficiently allocating the radio resources among users having different QoS demands. Group/cluster based MTC feature is one of the effective methods to serve the above purpose [65]. Scheduling prioritization is imposed on these clusters. In the following, methods to effectively arrange massive number of transmission requests with diverse QoS requirements into clusters of varying requirements are discussed. Also, different schedulers aware of versatile QoS demands of MTC devices are discussed.

Group channel access achieves significantly smaller access delay and signalling congestions than the conventional access scheme. Furthermore, it enables efficient resource reuse with less power and energy consumption. As data sent by MTC devices is aggregated by the controller/clusterhead and then transmitted to eNB, it reduces the transmission range and in turn reduces the transmit power of each MTC device. Also, as each device is scheduled

to send/receive at particular instant and need not to be awake all the time, it accounts for less energy consumption. Moreover, cluster heads are usually selected in a way that minimizes the total energy consumption and they may rotate among the MTC devices to balance energy consumption.

### **QoS-aware Fair Scheduling**

Scheduler for M2M communication should consider not only the throughput optimization but also the QoS differentiation in an effective manner. In [60], a downlink MAC scheduling algorithm is proposed that differentiates between different QoS classes and their requirements. Guaranteed bit rate (GBR) and non-guaranteed bit rate services are considered. It exploits different users' channel conditions and tries to create a balance between the QoS guarantees and the multi-user diversity in a proportionally fair manner.

Incoming packets are categorized based on their priority order into five different MAC QoS classes defined by the MAC scheduler. Authors used two schedulers, named as time domain (TD) and frequency domain (FD) schedulers. The TD scheduler is used to differentiate the users according to their QoS characteristics. While FD scheduler is responsible for assigning the RBs among the priority users. TD scheduler creates two candidate lists (one for GBR and the other for non-GBR). The FD scheduler starts with this GBR candidate list. The physical resource blocks (PRBs) allocation is done iteratively, where one PRB with the highest signal-to-interference plus noise ration(SINR) value is allocated for one bearer in each iteration. The bearers in the candidate list get orderly chance to select the next best PRB upon the highest SINR value. This PRB allocation process continues until all bearers in candidate list get one PRB. At the end of each iteration, the achieved data rate of each bearer is calculated and checked if sufficient data is available in the bearer buffer to be served. If sufficient data rate is achieved, then the bearer is removed from the candidate list for scheduling.

## Ubiquitous Massive Access

In [33], managing radio resources over clusters instead of individual MTC devices and the corresponding benefits are presented. In the proposed scheme, eNB manages RB on a cluster basis rather than on individual basis thus alleviating the complexity. Basic idea is the formation of MTC device (indexed by  $i = 1, 2, \dots, M$ ) groups or clusters and each cluster is associated with a prescribed QoS profile. Cluster formation is done on the basis of packet arrival rate ( $\gamma_i$ ) and maximum tolerable jitter ( $\delta_i$ ) for MTC devices.

MTC devices in the same cluster have identical QoS characteristics and requirements. A cluster with a larger  $\gamma_i$  has a higher priority. eNB allocates access grant time interval (AGTI) every  $1/\gamma_i$  ms for cluster  $i$  according to the priority. If AGTIs for different clusters are arranged in the same subframe, AGTI for the cluster with lower priority is postponed to the subsequent subframe. When a MTC device attempts to transmit data, it sends  $(\gamma_i, \delta_i)$  to eNB to request RBs. After checking the sufficient condition for guaranteed QoS, eNB will grant the service to the new device if the QoS (jitter) requirements of the new and all admitted devices can be satisfied. Thus, this condition is sufficient to ensure QoS requirements of all admitted MTC devices and in turn serve as access control.

Sophisticated calculations are avoided in the above proposed group/cluster based radio resource management. As a result, the computational complexity can be reduced when supporting a large number of MTC devices. Each cluster has different priority to occupy the shared medium, but only during the allowed AGTI. However, channel quality of each device is not taken into account in this work, which could have been used to increase the overall throughput. Although, this group based scheduling decreases the signalling traffic, QoS class identification raises several issues. Due to diverse traffic characteristics of MTC devices, the number and class of each application must be carefully chosen. Because, dividing the devices into a limited number of QoS classes may deteriorate the system performance. Therefore, dynamic formation of QoS classes in particular application scenario is more appropriate in M2M communication, particularly when device topology and individual characteristics are

not known apriori. But defining the new cluster every time, a MTC device with different QoS profile connecting to the network, is not efficient since this may lead to a large number of clusters.

### **Scheduling with Queue Awareness**

A packet scheduling scheme is proposed in [45] to cater for diverse QoS requirements in M2M networks. Authors proposed an analytical model for predicting the QoS performance of M2M services using fixed periodic scheduling (PS) algorithm. Afterwards, a modification to this scheme by exploiting queueing-dynamics is made. They extended the work done in [66], in which each terminal continuously reports to eNB about its buffer status and, at each TTI, eNB may grant any RB to any device. At each granted TTI, an intra-cluster prioritization scheme is employed, that is based on actual queue backlog of each device. Therefore, devices with larger queues are first granted access to the scheduled TTI.

This scheme enhances the delay performance of MTC devices, since random traffic dynamics are utilized by the scheduler when allocating the RBs to devices. However, this improvement induces a signaling and complexity cost, since the buffer status reports should be fed back to the eNB just before the cluster granted TTI begins. Also, decided RBs and scheduled pattern should be sent to the devices thus increasing the signalling overhead.

### **2.2.4 Efficient Spectrum Utilization**

To increase spectral efficiency, the same spectrum can be utilized for H2H and M2M communication. This mutual sharing of time and frequency resources among M2M and H2H communication can result in co-channel interference [61]. This interference plays a detrimental role in degrading the overall system performance. Therefore, it is important to examine efficient spectrum utilization for this purpose. Several types of performance indicators can be considered. It is important to make sure that all users are served with best channel conditions, and target of maximizing the number of users being served in particular time interval

should be met [55]. The utilization of radio spectrum for both domains can be improved by applying utility based assignment, as described below.

### **Resource Sharing Optimization**

Resource allocation and power control among H2H and MTC devices sharing the same resources are analyzed for different resource sharing modes in [62]. Minimum and maximum spectral efficiency restrictions, and maximum transmit power or energy limitation are considered for optimum performance. In the following, different modes of sharing the spectrum are discussed:

- **Non-Orthogonal Sharing:** Among cellular traffic and MTC devices using the same resources, causing interference to each other. eNB coordinates the transmit power for both links.
- **Orthogonal Sharing:** In which MTC devices are using dedicated resources. There is no interference between both types of users but resources allocated to them should be optimized.
- **Cellular Operation:** In which MTC device's traffic is relayed through eNB that acts as a relay node. It is assumed that the cellular network performs radio resource management functions for both the cellular and MTC connections. The portion of resources allocated to each user is to be optimized.

The eNB selects one out of the three possible allocation modes subject to existing channel and buffer status information. With non-orthogonal sharing, authors showed that the comparatively less number of optimal solutions exist than either of other two modes. Authors evaluated their scheme in a single cell scenario and Manhattan grid environment to show that substantial gain from effectively handling local traffic can be obtained. Thus, by proper resource management, MTC can effectively improve the total throughput without generating harmful interference to existing cellular network.

## Self Adaptive Persistent Scheme

A self-adaptive persistent contention scheme is presented in [63]. It deals with scheduling MTC devices in a periodic reporting manner for efficient spectrum utilization. Bandwidth utilization improvement along with congestion problem has been jointly dealt with here. This scheme takes advantage of the existing contention schemes to serve massive access of MTC devices, specifically when conventional access is unable to serve a large number of devices simultaneously.

The above scheme is comprised of contention phase and compact phase. In contention phase, attempt is made to decrease the collision probability, re-attempt count and access delay. In the compact phase, physical resource utilization for MTC devices is tried to minimize signaling overhead from core network to the front-end. MTC devices only contend on designated MTC device's physical random access channel (PRACH), which is not used by H2H. Therefore, H2H devices are not influenced by MTC device access traffic. Also, MTC devices remember the optimized contention information to achieve contention-free RACH thus minimizing the percentage of additional RBs used by MTC devices during random access. Also, every device can adjust itself using a dedicated contention resource to access the network; therefore, success probability, access delay and the number of preamble transmissions are optimized.

The above scheme increases the access probability and reduces the number of preamble transmissions of MTC devices. Furthermore, it mitigates the impact to H2H devices and optimizes the uplink bandwidth utilization of MTC devices. The access delay of the contention in the proposed scheme is longer than that in the legacy scheme at first, and then it improves.

## 2.3 Random Access

When a device switches on or comes out of idle mode to get connected to the network, the device does not have any resource or channel available to inform the network about its desire to connect to it, so it will send its request over a shared medium; this process is called random access, done via random access channel (RACH) in LTE. There are two situations of random access: (a) contention based random access, and (b) non-contention based random access. In contention based random access, there can be many devices in the same area/cell sending same request in which there is also a possibility of collision among the requests coming from other devices. In non-contention based random access, the network can inform the device to use some unique identity (e.g., unique random access code) to prevent its request from colliding with requests coming from other devices. On the other hand, non-contention based random access is used for few specific usages, e.g., during handover and resource request in connected mode operation non-contention based process to reduce handover and resource allocation latency so that the service interruption can be reduced. Non-contention based random access for initial access, especially for massive number of MTC devices is not practical because there is only a small number of random access codes unless one can over-provision the amount of random access resources. For contention based random access, network vendors and operators evaluate and decide on the amount of air-interface random access resources (PRBs) needed to have the required RACH success rate. With a low or moderate arrival rate of H2H connection setups, a reasonable amount of RACH resources per frame or per second is adequate for required RACH success rate. But, with the massive number of M2M devices, it is not straightforward how much of RACH resources are needed, especially when the arrival pattern of the M2M connection setups are not known. Typical RA operation comprises four steps as shown by Figure 2.3:

1. MTC device transmits a randomly selected RA preamble sequence (announced by eNB in the network) on physical random access channel (PRACH) to the eNB.

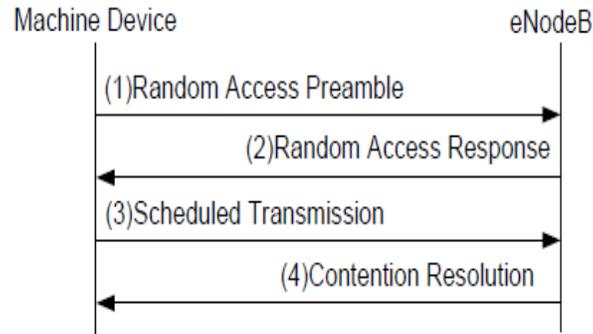


Figure 2.3: Random Access Steps [1]

2. eNB transmits an RA response on the physical downlink shared channel (PDSCH) in response to the detected preamble sequence. The response includes a preamble identifier, uplink grant, and time alignment (TA) information.
3. MTC device adjusts uplink transmission time according to TA time. Then it transmits its identity including RNTI and other messages (e.g., scheduling request) to the eNB using the physical uplink shared channel (PUSCH) resources assigned in the RA response in the second step.
4. eNB echoes the device identity it received in the third step on PDSCH.

**Potential Problem:** Problem will occur If numerous devices try to access simultaneously, in a situation when preambles/time-frequency resources are limited. It leads to a larger chance of devices choosing the same preamble in the first step of choosing access resources. If many devices send preambles with the same time/frequency resources, this will cause the eNB to get the same RA RNTIs resulting into same UL grants and preamble indexes. Therefore, all the M2M users(initially selected the same preamble) will send the data to eNB on similar resources. However, the eNB just decodes the first received message and It means other users will fail to access. If none of the users receive RAR message from eNB, they will back off with a random time. In overload situation, number of optional backoff values is far less than the number of contending MTC devices. These devices are choosing

the backoff values at the same time, so may end up choosing the similar value resulting again into collision in the next retrieval.

Detailed discussion on RA procedure can be found in [8, 67, 68]. In the following, we review some random access methods in view of MTC devices.

### **2.3.1 Random Access in MTC Devices**

Random access in M2M is analyzed in the literature in different context. Most of the research is based on resolving the contention among MTC devices when accessing the PRACH channel [69]. Few researchers have addressed the physical data channel (PDCH) overload detection and notification problem [15]. Few have made use of fixed timing alignment value between MTC device and eNB for static devices [1]. Others have presented the concept of contention based access (CBA) and discussed collision caused by it under high traffic load [70]. Slotted multiple access with collision avoidance (MACA) with traffic load adaptation is also done for M2M networks [71], where eNB on the basis of channel occupancy rate estimates the traffic and calculates the RTS transmission probability for MTC devices. In [72], an statistical approach for evaluation of synchronized traffic effects over signaling channels is presented. Authors concluded that synchronization expected in some MTC applications, can block the access channels or increase access delay. In the following, we will discuss few critical M2M problems, on which most of the contemporary research is based on M2M networks. Figure 2.4 categorizes few research directions for MTC device channel access.

### **2.3.2 Access Delay**

One of the important and challenging issues in M2M deployment, is to reduce the access delay in simultaneous channel access attempts from massive number of MTC devices. Both machine devices and mobile users may suffer from severe collisions during network entry/reentry, handover, scheduling request, and uplink timing synchronization. As a result, they have to backoff their transmissions. This backoff and resulting retry attempts further add to the

access delay. Therefore, efficient contention resolution with effective backoff algorithms are required for such contention based scenarios. A comparative study on these backoff algorithms with retry limit and access priority differentiation has been presented in [73].

### **Dynamic Access Control**

In [74], fast retransmission and dynamic RA algorithm is proposed to support massive M2M devices by reducing the access delay. Authors proposed an estimation algorithm for finding the access arrival rate in order to achieve the dynamic control of RA. In their scheme, MTC devices suffering from collision during RA initial steps, will send a preamble immediately in the next RA slot. Before each slot, eNB broadcasts dynamically calculated access rate.

Authors compare access delay and throughput of their scheme with backoff algorithm used in 802.11 DCF. It was shown that the maximum throughput of the proposed algorithm is higher even when the arrival rate of access is higher than the limitation of time slotted Aloha scheme ( $1/a$ ), while the throughput of backoff algorithm decreases rapidly when the average arrival rate is higher than ( $1/a$ ), where  $a$  is the arrival access rate. It is because, in the proposed algorithm, the number of accesses in each slot no longer follows Poisson distribution owing to the fast retransmission scheme.

Backoff scheme brings greater time delay and fails to deal with higher arrival rate of access, especially when there are large number of nodes. The proposed fast retransmission and dynamic access control algorithm, deals with the congestion in multi-channel RA under very high arrival rate of access. It is able to achieve a comparatively low delay effectively utilizing the channels. Therefore, it is a better choice in unique M2M environment for the efficient channel utilization.

### **Multiuser Detection**

To enable the efficient and low latency machine type communication, a contention based assignment method is proposed in [70]. MTC devices transmit packets on randomly selected resource without scheduling, which results in collisions. This problem is addressed by using

multiuser detection technique, in which radio network temporary identifier (RNTI) of the collided MTC device is decoded and used for regular scheduling in the subsequent subframe. As resources are allocated for a group of MTC devices, collisions may occur when multiple devices select the same resource. Authors suggested that RNTI sent by each device along with the data on the randomly selected resource, should be used to resolve this issue. Due to the small size of RNTI, it can be transmitted with most robust modulation and channel coding scheme and can be successfully decoded when transmitted on the same time-frequency resources. Afterwards, dedicated resources are allocated for data retransmissions bypassing the regular preamble selection procedure.

As MTC devices select resources randomly without indications from eNB, it saves signaling overhead. But this performance greatly depends on the coding rate of control information and number of receiving antennas used in a system. If number of devices increases, the interference is very severe which causes lots of retransmissions and hence latency is increased. With more antennas, the channel capacity is increased and hence retransmission is reduced. Multiuser detection is an expensive technique and its usage in everyday M2M applications is infeasible. But this work can be useful in sophisticated M2M applications in areas such as in military and surveillance.

### **No or Low Mobility**

Some work has been done in joint massive access control and resource allocation in mobile MTC devices. In contrast to mobile devices, fixed-location machine devices have a fixed uplink timing alignment (TA) value which is considered in the random access procedure. Based on this in [1], authors focus on fixed-location M2M services, such as smart metering and remote sensing, by considering metrics such as collision probability and access delay to evaluate their RA scheme. During initial access to a serving eNB, MTC devices use the conventional RA steps, acquire and store their TA values. Since then, MTC devices will utilize their own TA values assuming that it is fixed and unchanged unless the serving eNB is updated. In the next RA procedure, a machine device which receives a RA response

compares the TA value in the response with its own TA value. If it does not match, it performs a retransmission procedure by selecting a random backoff value and by waiting for the backoff time. If the received TA value matches with its own TA value, it continues to perform a scheduled transmission.

The RA scheme discussed above is based on fixed information at a large number of fixed devices, thus signaling overhead for location update is not necessary. Their proposed scheme reduces collision probability, lowers average access delay, and achieves energy efficiency compared with the conventional RA scheme. It enables the fixed-location MTC devices to predetermine whether they can advance to the next step for scheduled transmission with a higher probability of no collisions. Despite the above mentioned improvements, their scheme suffers from unfairness issues depending upon the distribution of MTC devices. In a uniformly distributed case, as the distance from the eNB increases, the number of devices with the same TA also increases. Thus, the group which is nearest to the eNB has a lower collision probability than the group which is farthest from the eNB.

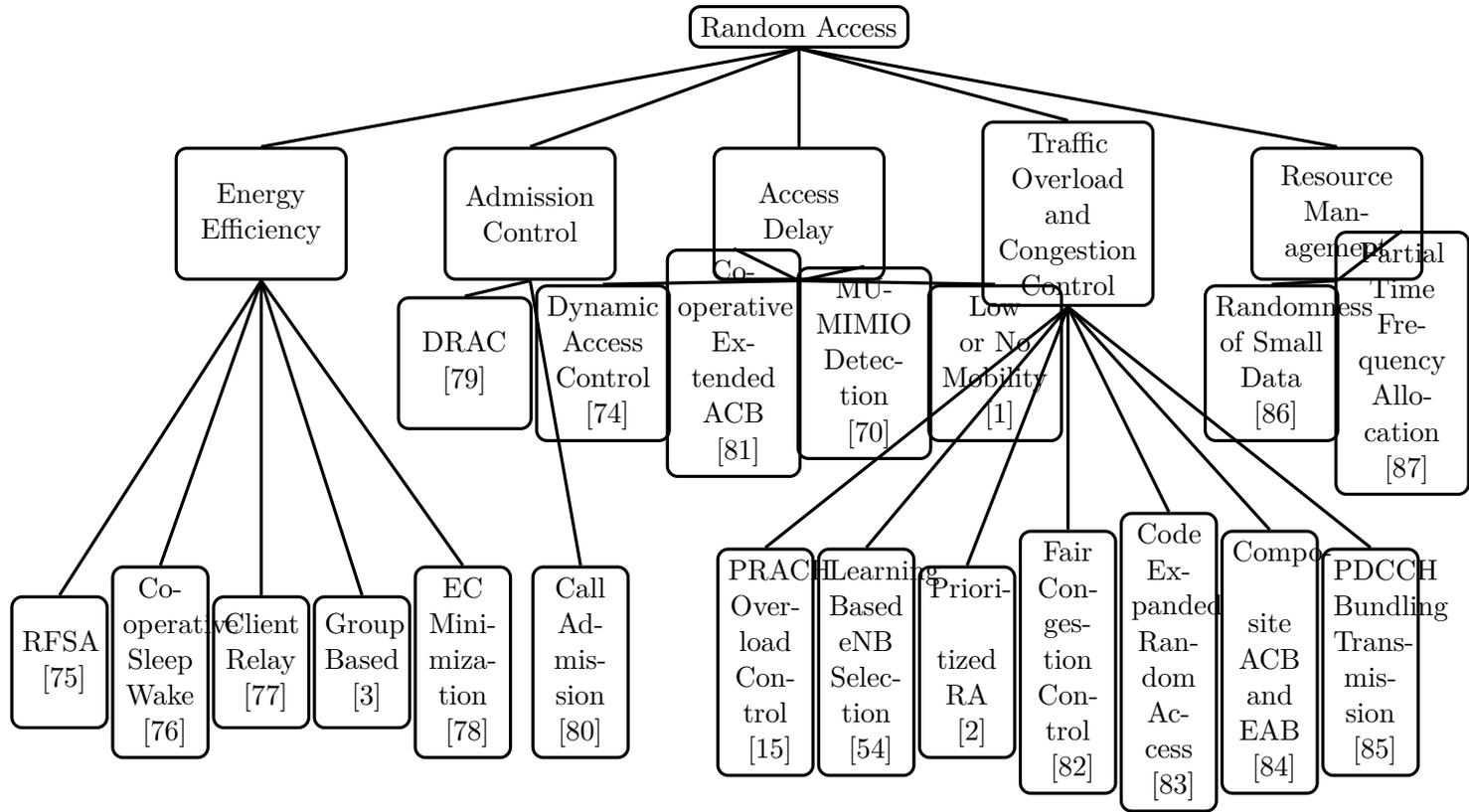


Figure 2.4: Random Access Objective Types

## Cooperative and Extended Access Class Barring

Authors in [81] presented the idea of cooperative access class barring (ACB) for access load sharing to improve the access delay and congestion in M2M networks. Though the cooperation among eNBs is already available in practice, they reinforced it by proposing cooperative ACB. Evaluation was focused on two classes of performance metrics, (i) first the average (access) delay and the average throughput averaged over all MTC devices and (ii) secondly, the worst (access) delay and the worst case throughput that is the largest delay among all MTC devices. In the same spirit, evolutionary game based ACB scheme is proposed in [88]. Where MTC devices and all ACB parameters of eNBs are jointly optimized, based on levels of congestions, resulting in effective improvements in access delay.

Extended access barring (EAB) has been recently introduced in 3GPP to control the access load. Access class barring is used for H2H devices to provide prioritized access for QoS users. In [89,90], authors proposed EAB for MTC device overload control and showed that overload situation can be effectively resolved. They gave complete model for the signaling and overload control, by first calculating the theoretical maximum admission rate the network can handle, and then EAB parameters were tuned accordingly to provide that admission rate.

### 2.3.3 Traffic Overload and Congestion Control

One of the main challenges in M2M communication, is to efficiently handle MTC traffic characteristics, specifically the load generated by massive simultaneous low data rate transmissions. Therefore, network should have the capability of counteracting signaling and traffic load spikes caused by a sudden surge of massive numbers of MTC devices trying to access the eNB all at once. This massive access can be due to several devices requesting to transmit at the same time by some applications or due to roaming MTC devices suddenly attaching due to a fault in their home network thus causing overload. In such situations, radio and signalling network congestions may occur due to massive concurrent transmissions which

can lead to large delays, packet loss and, in the extreme case, service unavailability. Various overload control techniques are defined by 3GPP such as M2M server triggering eNB for paging M2M terminals, broadcast-based, rejecting connection requests in the radio access network and in the core network etc.

This increased overload primarily affects the PRACH and, is further aggravated if M2M terminals try to repeat their access attempts without realizing that the unsuccessful RA attempts are due to PRACH overload. Several solutions have been proposed to deal with this overload situation such as access class barring [89], priority based backoff scheme [83,91], methods for separating RACH resources [2], and increasing the RACH resource dynamically to handle the overload [92].

### **Overload Analysis**

When a large number of MTC devices attempts to access the channels thus causing the network overload, the loading effect of non-active and active devices is evaluated in [93]. In the overload situation, MTC devices should enter detached state if they do not have data to transmit. However, the drawback of a non-active MTC device staying in idle mode is that serving eNB has to keep some information of the device for further use. Authors compared two networks, one with active devices and another with nonactive devices. They concluded that the loading of an eNB to handle idle and detached MTC devices is quite similar. Considering the power saving and the complexity of context storing and forwarding, MTC devices shall enter detached state if they do not have data to transmit.

### **PRACH Overload Control**

An important issue of overload control is PRACH overload detection and notification, and it has been discussed in [15]. Current LTE-advanced RA procedure includes a simple algorithm for adjusting the PRACH transmit power for each unsuccessful RA attempt. The other resources are not adapted according to the PRACH channel load condition. Here, authors have devised a mechanism for eNB to automatically add or reduce PRACH resources when

it detects an increase or decrease in PRACH load, respectively. A M2M terminal can be instructed by the eNB to back off for a period of time before repeating the RA attempt. Even though this backoff mechanism can be employed to reduce PRACH channel overload, it is not scalable because all the M2M terminals do random backoff .

### **Reinforcement Learning Based eNB Selection**

In typical overload control schemes, eNB selection for load sharing is not considered, which is essential for efficient RA based M2M communication. To avoid congestion caused by random channel access of MTC devices, a reinforcement learning-based eNB selection algorithm is presented in [54]. It allows the MTC devices to choose the eNBs to transmit packets in a self-organizing fashion. A MTC device in an overlapping area of multiple eNBs can choose the eNB that maximizes its QoS performance. In this case, the MTC devices will observe, learn, and adapt the eNB selection decision independently. Reinforcement learning algorithm is used to address the eNB selection criteria, and specifically used as a load balancing method to distribute MTC devices among the available eNBs. With the proposed eNB selection algorithm, MTC devices have the ability to switch to the eNB that provides better performance (i.e., smaller delay). In this thesis, reinforcement learning is used to allocate time slots among competing MTCs.

### **Prioritized Random Access**

A prioritized random access (PRA) scheme is proposed in [2] to efficiently solve the RAN overload problem and provide QoS for different classes of MTC devices. This is achieved by pre-allocating RACH resources for different MTC classes with class-dependent backoff procedures and preventing a large number of simultaneous RACH attempts by using dynamic access barring (DAB). The authors have classified the MTC devices into categories. They pre-assigned different amount of virtual resources for different MTC classes as shown in Figure 2.5. The eNB decides the virtual resource allocation according to the knowledge of the statistics of the MTC devices. Results shows that this scheme perform better than EAB

schemes in terms of access probability and average access delay.

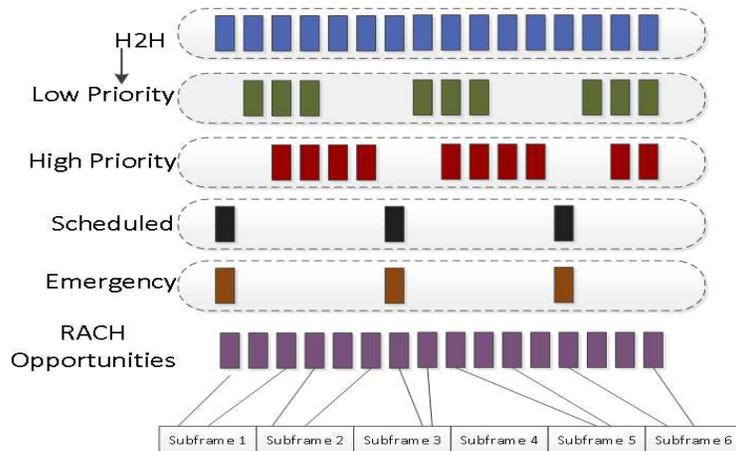


Figure 2.5: Virtual Resource Allocation For MTC Classes [2]

### Code Expanded Random Access

In [83], authors presented a scheme to sustain a large and bursty random access of massive MTC devices. Their scheme increases the amount of available contention resources without the increase of contention subframes or preambles. This increase is accomplished by expanding the contention space to the code domain through the creation of RA codewords. Each device selects a codeword consisting of a randomly chosen preamble per every subframe of the virtual frame. In this way, the number of contention resources is expanded and the amount of collisions is reduced. But the collision occurs when two or more MTC devices select the same codeword. Authors complimented this method with adaptive EAB and creation of user classes, that were predicted according to the expected loads and traffic types.

In the above schemes, by selecting the appropriate number of RA codewords, it is possible to maintain the RA scheme efficiency over a large load region. For the same number of RA access subframes and orthogonal preambles, the amount of available contention resources is drastically increased, enabling the support of an increased number of MTC devices.

## Composite ACB and EAB

In [84], ACB and extended access barring (EAB) based on M2M traffic models are presented as an overload control mechanism for MTC devices. Authors divided all UEs in different access classes (ACs) and allow only specific class at a time depending upon barring factor, which is the probability that it will be allowed or not.

- ACB: When network experiences overload, one or more ACs can be allowed while others are barred from accessing the network. By suitably changing the barring factor and barring time, network can control the proportion of allowed MTC devices accessing PRACH and also their re-attempt rate. Thus it is very effective in controlling the excessive access requests and signalling overhead. Also, it is helpful in improving the access success rate significantly without introducing much penalty in terms of access delay. One of the drawbacks of ACB is that, both the barring factor and barring time are common for ACs. There is no mechanism to differentiate the service quality in terms of barring probabilities for these classes of MTC devices. EAB has been introduced with ACB to overcome this drawback.
- EAB: It is a method to control access overload and signalling congestion in M2M, by allowing only a subgroup of devices at a time, so that collision among themselves is manageable. One of the distinguishing factors of EAB from ACB is that in EAB the ACs are either completely barred or unbarred, whereas in ACB it is based on the probability. This means in EAB, only devices belonging to unbarred ACs can attempt network access, whereas in ACB, among the devices belonging to the same AC, some devices are barred while others are not. It is noted that MTC devices which is configured for EAB, will first check EAB and than ACB to access PRACH.

## Fair Congestion Control

In [82], fair congestion control method for terminal groups for telemetering and disaster recovery, is proposed. This method distinguishes between RA collisions and idle signals using

MAC layer information independent of physical layer specifications. The proposed method is based on the existing exponential backoff method, the ALOHA reservation method and uses exclusive and shared time slots (TSs). The eNB compares the number of received RAs at both TSs, detects RA overload, and then controls the RA rates and packet rates of the terminal groups.

Authors assumed that only important services construct exclusive groups thus using exclusive TSs while other services share a slot by sharing a group. Some terms to understand their method are listed below.

- Available Packet Rate (APR): of a group is defined as the sum of the packet rate that is generated in terminals of the group and received by the base station.
- Offered Packet Rate (OPR): of a group is defined as the sum of the generated packet rate in terminals of the group.
- Minimum Packet Rate (MPR): for each group is a static value and if the group's OPR is equal to or less than its MPR, the APR shall be the OPR. Here, the sum of MPRs shall not exceed 1. The MPR shall be able to be set to any value between 0 and 1.

If the group's OPR is more than its MPR, those groups share the remaining bandwidth, and the APR of each group shall be proportionate to its MPR. The fair congestion control method for terminal groups shall be effective at any OPR per terminal between 0 to 1, even if the OPR changes dynamically.

With the proposed method, minimum call rate is assured to the terminal groups for specific services even if all terminals make simultaneous RA attempts and re-attempts. This method uses only MAC layer information, so it can be combined with any physical layer specifications. In addition, eNB was able to control the packet rate of each group including all the terminals persistently attempting to connect.

## **PDCCH Bundling Transmission**

In [85], authors dealt with congestion problem on the physical downlink control channel (PDCCH). They provide mathematical analysis of access delay by using queuing theory. Authors also evaluated access performance of M2M through system level simulation and showed that the congestion on the air interface is very serious in some M2M applications. To ease the congestion, they proposed bundling transmission scheme over PDCCH. They suggested to choose one value from the reserved RNTI and define it as the MTC-RNTI. eNB multiplexes the grant information of multiple MTC devices into one packet data unit (PDU) and identification of each multiplexed MTC is masked with MTC RNTI. MTC devices monitor PDCCH channel for both RNTI, i.e, for their cell and MTC RNTI. After receiving the PDU, MTC device searches for their own grant information using their cell RNTI.

Two application scenarios of normal and emergency services are considered in [85]. The normal case includes applications in which the access management can be applied and MTC devices obtain access to the network in coordinated manner with uniform distribution. While for emergency case, MTC devices access the network in an uncoordinated manner and beta distribution is used to model the arrival rate. It was shown that almost 90 percent of MTC devices finish the access within 100 ms by using proposed bundling scheme while the percentage is only 60 percent without bundling transmission.

## **2.4 Resource Management**

### **2.4.1 Resource Allocation**

Efficient, fair and less complex resource allocation is important in M2M networks with distributed implementation for a large number of MTC devices. Fixed channel allocation though easy to implement often cannot satisfy the unbalanced and dynamic traffic demands in M2M networks [10]. M2M communication, which can utilize cellular networks as underlay and reuse the cellular resources, can potentially achieve improved spectral utilization

and cell throughput enhancement. With optimum resource allocation for MTC devices, M2M communication can effectively improve the total throughput along with interference management between MTC devices and traditional cellular devices [94]. Orthogonal resource allocation between two kinds of links is one of the options to have no interference among them. This sharing method is simple to implement; however, the resource efficiency still has room to improve when using non-orthogonal sharing. Although with this sharing, high spectrum efficiency can be achieved, interference exists between M2M and H2H users, preventing the system throughput from being further enhanced. Therefore, to mitigate interference is another critical problem to deal with. Coordinated multipoint transmission and reception (CoMP), in particular joint processing and coordinated beamforming, has proven to be a beneficial solution for interference management in cellular networks [95]. Hence CoMP, where multiple eNBs form a cluster and cooperate by exchanging signaling or user data via the core and backhaul networks, can be utilized for interference mitigation in MTC devices as well. In the following, various resource management methods with different perspectives for M2M communications are discussed.

To address the interference problem in M2M/H2H communication, an intelligent resource allocation scheme for massive number of devices is presented in [87]. There, MTC devices share only a part of each cellular user resources, such that the interference on regular cellular communications can be suppressed. In the mean time, MTC devices can share multiple cellular users' resources to meet their own QoS requirements. This is unlike a conventional resource allocation scheme where each device shares only one cellular user's entire resources. Authors formulated the problem of radio resource allocation for M2M communications as a mixed integer nonlinear programming, and proposed an greedy heuristic algorithm to lessen interference utilizing channel gain information. They evaluated the network performance through three parameters: system capacity, the stability of H2H communication and MTC service rate. Stability of H2H communications is reflected through MTC impact on cellular users. Their proposed scheme enhances the overall system capacity and provides better system stability in M2M communications underlying cellular networks while guaranteeing

the performance of M2M communications. As M2M pairs share a part of cellular users resources, M2M interference to one specific cellular user would be weakened in this design.

One of the most important problems in M2M networks is to efficiently handle the heavy random access (RA) load caused by a large number of MTC devices. It includes RA preamble allocation management followed by efficient resource allocation and management. Due to the unique characteristics of M2M and related emerging applications, RA load is expected to be more challenging. Increase in preamble contention results in increased collisions and in turn increases the retransmission attempts. The number of transmission attempts and a timeout period for uniform backoff after a collision, are the measures to quantify the throughput and efficiency of the distributed networks [96]. In [86], RA preamble resource management method to accommodate new network traffic patterns caused by frequent transmissions of small data is presented. Authors presented two methods for RA preamble allocation and management. First method is to split the set of available RA preambles into two disjoint subsets: one for ordinary data applications and other for small data applications. While in the second method, second subset is for both types of applications keeping the rest same. Authors did throughput comparison between the two types of methods and demonstrated that each method has benefits in throughput over the other depending on the load conditions.

### **2.4.2 Energy-Efficient Algorithms**

M2M network usually consists of a large number of typically power constrained devices, such as battery equipped sensors, that gather data and relay information without human interaction. Thus in order to prolong network lifetime and provide "green" communication, M2M networks should take into account the energy efficiency requirement. In many M2M applications, saving energy for machines is more imperative than increasing the throughput, because machines usually transmit small data but have limited energy. One of the critical issues of M2M is how to deal with a large number of access requests from massive amount of devices while maintaining low power consumption of the embedded M2M system with

tolerable latency.

There are four main sources of energy consumption in M2M networks: energy required to keep the communication radios on; energy required for the transmission and reception of control packets; energy required to keep sensors on; and energy required for data transmission and reception. The fraction of total energy consumption for sensing remains constant and cannot be controlled. While energy usage for data transmission/reception is optimized by data aggregation schemes and hence, the energy expended to keep the communication system on is the dominant component of energy consumption. In literature, there are several ways for minimizing the energy consumption such as minimizing collisions, retransmissions and idle times, and clusterings. A most common approach is duty-cycling the radio components of the devices i.e, devices can switch on and off their radio transceivers to avoid waste of energy when no communications are being carried out. Duration of the duty cycle period will determine how efficiently devices can communicate [97]. Some of the work done in this respect are discussed next.

### **Cooperative Sleep and Wake**

An effective way to minimize energy consumption and to prolong network lifetime is to reduce the time spent by nodes in idle listening. This idle listening consumes a large amount of energy and various sleep-wake scheduling protocols have been proposed in literature [98, 99] to deal with them. In these papers, authors have considered a source communicating with the corresponding destination with the help of multiple relays [76]. Authors also determine the optimal transmission energy at each active relay, while providing QoS assurance to the receiver. Instead of selecting a single relay and assuming fixed energy transmissions [98], they combined cooperative communication techniques and sleep-wake mechanisms. Multiple relays are selected assuming non fixed energy transmissions and also practicing sleep periods of fixed length. In order to determine active relays, three alternative relay selection criteria are used. The energy of the active relays is computed according to the minimization of the total energy, subject to meeting error performance constraints. The proposed energy efficient

sleep-wake mechanism combining relay selection techniques and sleep wake mechanism takes place in two phases:

- In the first phase,  $N$  relays are selected to participate in the transmission process out of  $L$  relays using three alternative criteria. These include, best channel state information or the largest residual energies or a composite criterion involving both residual energies and the channel state conditions of the relays. The remaining  $L - N$  relays that are not selected sleep for a fixed time period.
- During the second phase, optimal energy allocated to the  $N$  selected relays is computed, simultaneously guaranteeing a certain required QoS level for the destination node. The computation of the energy allocated to each one of the  $N$  selected relays is based on the minimization of the total consumed energy at the relays, subject to meeting a specific constraint on the received SNR.

The method discussed above is shown to increase network lifetime and achieve a balanced tradeoff between energy consumption and error performance.

### **Group Based Selection**

In the context of network resource savings, authors in [3] suggested group based transmission. They proposed MTC access control algorithm to ensure low uplink energy consumption (EC) of MTC. MTC devices are arranged into groups. After grouping, one device will be assigned as the coordinator in each group, and only coordinators are allowed to communicate with eNB as shown in Fig. 2.6. This grouping and coordinator selection scheme result in devices accessing with reduced uplink EC of the MTC system. Moreover, implementing grouping reduces the redundant signaling to avoid congestion. In Figure 2.6, Link 1 is between MTC device and coordinator and Link 2 is between coordinator and eNB. Both links operate in different frequency bands to avoid interference. Moreover, weighting factor which takes channel condition between the coordinator and eNB into consideration, is introduced to further improve EC performance. A machine learning algorithm which clusters data rapidly

with low complexity, is applied for the group formation and coordinator selection in each group. Optimum energy consumption is implemented by exhaustive search to minimize the energy consumption in each group. Various considerations of channel condition between coordinator and eNB are also given.

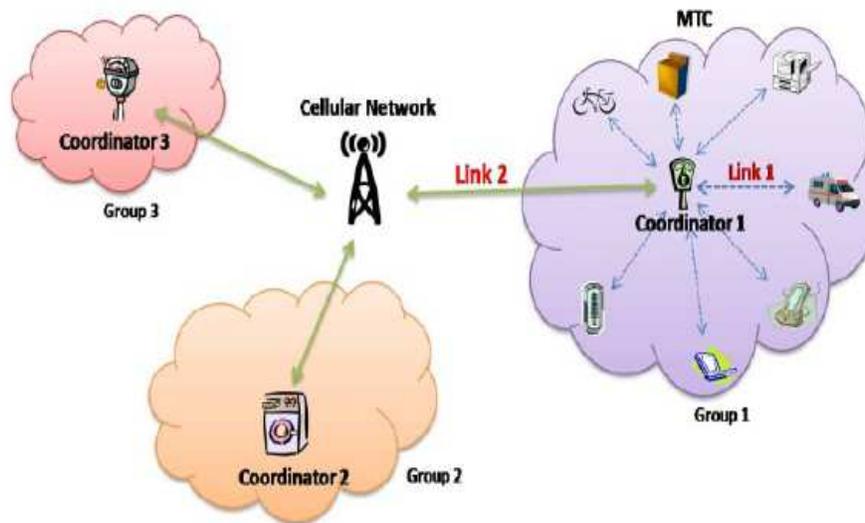


Figure 2.6: Coordinator Random Access [3].

Similar grouping scheme is also used in [14], where joint massive access control and resource allocation schemes are proposed. It performs machine node grouping and coordinator selection, and coordinates random access with a consideration of minimizing transmission and circuit energy. Also, they have devised a mechanism to determine the proper number of groups under a 2-hop transmission protocol, with minimum energy consumption in both flat and frequency-selective fading channels.

Group based MAC can reduce signaling overhead and simplify the traffic, policing and charging, thus increasing the energy efficiency of MTC devices. However, formation of groups is done for minimizing the energy consumption without taking into account the delay requirements of the devices. In the proposed scheme, group leader accumulates the received uplink traffic and periodically merges the messages at its own internal buffer. Therefore,

a group leader having a large internal memory and good communication link quality is required. Selection of a group leader is also another problem to be further researched.

## **Data Aggregation**

Applying data aggregation is one way to reduce energy consumption by reducing transmission overhead and prolonging the lifetime of M2M network. Proceeding with the same idea, various data aggregation schemes are proposed in [13, 100–102]. In [101], two methods based on the Karhunen Loe (KLT) or partial KLT transform for data aggregation in M2M gateways are proposed. Main focus is on the use of capillary M2M networks for the estimation of spatial random fields. Capillary network is composed of a high number of devices (e.g., sensors) equipped with short-range radio interfaces. This capillary network is coordinated by a M2M gateway which is responsible for collecting spatially correlated sensor observations and aggregating them prior to their re-transmissions to a remote application server. Optimal duration of these two transmission phases for providing best trade-off in terms of packet collisions and compression level is determined. Also, the effect on optimal point with the change in spatial correlation properties of the underlying field is investigated.

To utilize data aggregation, M2M devices need to have message buffering by utilizing a buffering timer. Upon its expiration, a M2M device retrieves data collected in the buffering period and aggregates them into a new message, which increases the transmission delay from the source to the application server. To address this issue, authors proposed an analytical model in [102] to investigate the configuration on buffering. Authors discussed the performance metrics related to latency of the application server, aggregated volume and energy consumption against the buffering time configured in MTC devices.

Data accuracy, latency, energy efficiency and network lifetime are four key factors used to determine the efficiency of data aggregation techniques. Data aggregation is a solution to increase the efficiency and decrease the overhead associated with small and frequent M2M data transmission. However, it poses extra challenges on per-hop and end-to-end security since aggregating nodes essentially need to access the information content. Due to the fact

that wireless medium is lossy in nature, longer aggregated packets are at more risk of errors than shorter non aggregated messages. Also, long non intended packets can be received from a valid node causing exhaustion of the nodes' battery.

### **Client Relay Scheme**

For reliable data transmission, energy expenditure exponentially grows with distance. Hence, relaying the data over shorter intermediate hops seems to be beneficial. In this context, a client relay scheme is proposed in [77] to improve the link reliability and energy efficiency for MTC devices. Authors evaluated this scheme across client throughput, latency, and energy consumption. They pointed out that latency and energy expenditure of cell edge M2M devices may be dramatically lowered, when there are simultaneous network entry attempts by a large number of MTC devices. Such surge in network access attempts may occur, for example, in a power outage scenario where a large number of smart meters attempts to connect to the network to report the outage event, and again when they reconnect to the network upon restoration of power. Their proposed client relay scheme can help ensure that the performance of other cellular devices is not adversely impacted by a large number of uncontrolled network access attempts from M2M devices.

In the same context, authors in [103] have discussed the issue of relay selection, from a pair or multiple pairs of MTC devices, serving as two-way relays for the cellular links. Since MTC devices and cellular user may potentially be at different locations, their channels towards eNB and cellular users are different. Therefore, when different MTC devices act as the relay, they can help the cellular links achieve different rates. Relay selection criterion depends on the channel state indicator (CSI) of both two-way relays, and MTC device links unlike conventional relay channel where CSI of only two-way relay links are generally considered.

Client relay is believed to be a promising concept that can boost the performance of wireless cellular networks. By reducing transmission power through the use of client relays will also reduce the overall interference to the neighboring cells. Therefore, these techniques

may be preferred under interference limited conditions. Also, cell edge user performance is improved when devices collaborate with each other. However, finding the optimal distance to locate a relay between a source and a destination for a reliable transmission is a challenge [104].

## **RFSA**

In [75], reservation frame slotted aloha (RFSA) is recommended for the channel access of MTC devices. Authors evaluated its performance through simulations and showed that it is more efficient than frame slotted aloha (FSA) [53], as it guarantees collision free data transmission once device succeeds for the first time. Performance of RFSA is evaluated under M2M data burst transmission case, assuming delta traffic model. Authors assumed a group of devices switching off their radio interfaces for certain periods of time to save energy. They wake up periodically and transmit data, thus abruptly setting network to the saturation condition. Each device intending to send  $L$  packets is waiting for the coordinator to send a request for data. Afterwards it transmits a packet to the coordinator without carrier sensing. If the coordinator has received the first data packet correctly, the successful device reserves that slot for next  $L - 1$  frames. After sending complete packets, device will release its slot for other devices in the subsequent frames. This is unlike FSA, in which the device randomly selects one of the slots for each packet, thus contends independently for each packet of the burst.

Authors showed that under idle/saturation conditions, RFSA outperforms FSA, in terms of average contention period, network throughput, and average energy gains for the coordinator and the devices. They concluded that there exists an optimal frame length which optimizes the performance of the entire network for both the protocols. Proposed method is energy efficient as energy gain on the devices can reach up to 48 percent for large networks, and increases further with the increase in the size. Such energy gains accounts for the prolonged lifetime of M2M networks.

## Access Delay Constraint

Tradeoff between energy consumption and delay is investigated in [78] with respect to given resources. Flat and slow fading channel environment is considered assuming that static devices are distributed in rural area transmitting with low and fixed transmit power. As MTC device is characterized by low transmit power, each device can have at most 2-hop transmission, i.e., between MTC device and coordinator, and between coordinator and eNB. Authors aim to minimize the total consumed energy while satisfying the assigned resource and system delay constraints. First step starts by properly arranging MTC devices into groups followed by coordinators selection. After that, each group re-selects a proper MTC device as the coordinator to relay packets of other MTC devices in the same group to eNB. The two steps execute alternately until the result of group and coordinator selection remains unchanged. One of the most important features to maintain low energy consumption is to extend the battery life of MTC devices. In the given scheme, total energy consumption is minimized by reducing the energy required by each packet transmission.

### 2.4.3 Admission and Rate Control

Admission and rate control are crucial in efficient functioning of large number MTC devices in M2M networks. Its intelligent design helps to allocate radio bandwidth based on QoS demands in resource constrained M2M networks. Due to the versatile nature of MTC devices and their application specific data traffic patterns, automatic admission control is essential. An adaptive admission control should analyze and predict the QoS performance of devices, and accordingly allocate appropriate wireless bandwidth to accommodate more MTC sessions without QoS degradation. Several admission control schemes for M2M networks presented in literature are discussed next.

In [79], a cross-layer design of distributed admission and rate control is presented. A game theory analysis module (GAM) is used to model the competition of radio bandwidth among MTC devices. In addition, distributed QoS-aware and fair sharing for transmission

opportunities are provided. Each device contains media format management entity and the QoS management entity. Media management entity will examine that the requested session by a device matches its capabilities, and QoS management entity will decide whether the new media flow is acceptable or not. This decision is based on the results from two of its modules i.e., of queuing analysis module (QAM) and GAM. GAM will select and announce a tentative rate which could maximize the utility of the new media flow. Based on this rate, each existing media service will invoke its GAM and feedback with an updated rate. From then onward, all the media flows will compete under a non-cooperative game theory framework.

Admission control schemes proposed above suffers from higher call blocking probability due to poor handling of call requests. This is specifically true when the duration of transmission interval for an MTC device is longer than its delay constraint. To overcome such a limitation, an admission control scheme is proposed in [80] that improves the call blocking probability and reduces the call admission overhead by using the delay based priority assignment scheme. The proposed method manages the MTC devices into group, and set of devices with the same QoS requirement having QoS parameter is grouped together as one cluster. MTC devices communicate with MTC controllers through eNB and once admitted, they are allocated with radio resources for data transmission. Also it is assumed that there are delay constraints. The elapsed time from the data creation of an device until it finishes its transmission to an eNB is bounded by a maximum allowable delay. The eNB allocates the granted time interval (GTI) to the clusters, which is a set of the RBs having both time and frequency elements. A MTC device requests call admission by sending a request with its QoS parameter to eNB through the random access channel. If there already exists a cluster satisfying the QoS requirements, then the associated GTI has at least one available slot, the request is accepted and the MTC device joins the cluster. Otherwise, a new cluster is created. Also, it is checked in advance that the creation of the new cluster does not affect the delay satisfaction of the existing clusters. If the GTIs for different clusters are overlapped at the same time, the cluster that has a shorter delay is given a higher priority.

## 2.5 Prospective Scheduling and Random Access Techniques

It should be noted that 3GPP has proposed some important solutions for handling small data transmissions in M2M networks over LTE [42]. Considering the traffic generated by machine-type applications, RAN's ability is enhanced by reducing signalling overhead between MTC device and RAN, and also device's power consumption is reduced.

These are achieved via improved connection management for short-lived connections and improved handling of small data during live connections. Fewer control messages are proposed by combining the information exchange between MTC devices and RAN. During the idle-connected mode, the signalling is reduced by transferring small amount of data. Connectionless data transmission is proposed for uplink data in device's idle state using RACH procedure. Overall, to handle small data transfer in M2M networks, some of the mentioned solutions will require new procedures for random access and connection management in future M2M enabled LTE networks.

The proposed solutions for power optimization include extending DRX (discontinuous reception) value in idle modes applicable for delay tolerant applications. The MTC devices can assist in adjusting the DRX values. In the connected mode, longer DRX cycles are considered that will help devices to switch off their transceivers for longer periods of time. Another solution is to delay the transmission if the radio coverage and channel conditions are weak, again applicable for delay tolerant applications expected in traditional M2M networks. By introducing another state (besides connected and idle states), MTC devices are expected to save power. The above mechanisms will incur longer access delays and hence applicable for machine-type services that can tolerate traffic with longer access delays. The radio access protocols will require modification to accommodate DRX cycle extension. The level of burstiness in the M2M traffic profile will largely influence the savings that can be realizable using the proposed enhancements.

## Scheduling for Multi Class Services

Some authors have suggested scheduling and resource allocation schemes to deal with the QoS requirements of multi class system. They used system capacity and the number of effectively served requests as performance metrics. The resource allocation algorithm adapts dynamically to the number of requests in the system, assigning resources with as much fairness as possible.

Two scheduling algorithms are proposed: Single Channel Scheduling Algorithm (SC-SA) and Multiple Channel Scheduling Algorithm (MC-SA) in the context of multiple class traffic. SC-SA heuristic allows the assignment of at most one channel block per request and per transmission time interval (TTI), independently of its QoS requirements. The aim of the SC-SA allocation scheme is to allocate more radio blocks (RBs) to the requests experiencing the worst conditions, than the ones which have a higher performance. The MC-SA algorithm consists of allocating more than one resource block to the requests that are not meeting the throughput target. The reason for allocating multiple RBs to a single request is in order to help the requests such that the average throughput transmitted within one TTI is smaller than the throughput target. If the number of requests is higher than the total number of available RBs, the scheduler takes the requests that are experiencing the poorest performance and assign them a RB. Their suggested algorithms seem to be excellent choice for the mixed nature traffic of MTC devices, in which resource allocation assignment is done in a fair way such that the throughput and delay are adaptively adjusted according to the traffic load and traffic type.

## Semi-Persistent Scheduling

A large number of MTC devices imposes serious constraints to the applicability of highly dynamic scheduling approaches. It should support a massive number of MTC devices that generate small amounts of periodic/non-periodic data traffic. Due to the special traffic requirement of MTC, semi-persistent scheduling is a possible candidate in M2M networks.

It allocates an uplink traffic channel periodically without any additional control message during a traffic burst. An initial uplink traffic channel is obtained either by random access or via an assigned control channel, over which a periodic reservation is made. The eNB allocates an uplink grant that allows the terminal to send a buffer status reporting message, and finally a traffic channel is granted to the terminal. Semi-persistent scheduling makes an allocation decision for a longer time period; thus, it is not necessary to inform the MTC device on a TTI basis. This avoids the unnecessary signalling overhead which is a serious matter in simultaneous massive access attempts of MTC devices.

## **Subframe Allocation and Management**

RACH is responsible for achieving uplink time synchronization for a user which either has not yet acquired, or has lost its uplink synchronization. When multiple users transmit the same RA preamble code in the same RACH subframe, it will result in collision and cause delayed access by retransmission. For conventional RA procedures, some authors have suggested increasing the RACH channels for high volume of users. It is recommended to use the same scheme in massive M2M scenario. By increasing RACH subframes will also result in decrease of average delay. There exists a tradeoff between the RACH subframes and subframes for data transmission. Therefore, it is crucial to determine the number of RACH subframes as per the arrival rate, in order to optimize the overall network performance while satisfying a given delay requirement. A network should be able to detect the change of the arrival rate of the random access preambles, and adapt the number of RACH subframes accordingly.

## **Application Specific QoS Dependent Scheduling**

Diverse nature of M2M communication imposes different requirements on scheduler in contrast to ordinary systems. Ordinary LTE schedulers are mostly designed to maximize the throughput while taking into account some fairness and QoS rules. Also, due to the limited number of differentiated services (voice, video etc) and relatively very small difference in

their delay tolerance constraints, channel quality plays a critical role in scheduler characterization. In machine type communication, channel quality and QoS constraints are equally important for the design of a scheduling algorithm. It is also important to know that which device can tolerate the denial of service and for how long can stay in channel congestion situation. Knowledge of the exact delay tolerance of devices significantly affects the number of served devices. Therefore, an efficient M2M scheduler is required, which is able to continuously allocate shared channel resources among both types of users, maintaining required performance gains under diverse network conditions.

The emerging field of M2M communication has introduced new challenges for the communication and networking research communities. It has also created new multi-disciplinary research opportunities and potentials for collaborations among researchers in the communications, networks, signal processing, security, and sensor areas.

## **Other Research Possibilities**

Most of the focus of the current research is on traffic overload, congestion control, admission control and spectrum utilization. However, little work has been done in energy efficiency, energy efficient MTC device design, QoS provision support for massive M2M communication

Unique features of M2M communication such as the larger number of connected devices and the diversity of applications require specific enhancements to the existing communication system. Fixed cost for M2M communication is very high as M2M nodes are transmitting mostly small size data but the frequency of their making data connections is higher than H2H communication. This is due to their specific roles and functions: e.g., frequently reported small size measurement data.

One of the most important tasks is the energy efficient handling of massive RA load generated by a possibly huge M2M population and avoiding the resulting congestion. The energy required to sense events is usually a constant and cannot be controlled. While the energy expended to keep the communication system on (for listening to the medium and for

control packets) is the dominant component of energy consumption, which can be controlled to extend the network lifetime. Periodic, asynchronous, on-demand sleep wake scheduling are the most common ways to minimize energy consumption. However, additional delays are incurred to exchange synchronization information among MTC devices and may be unacceptable for delay sensitive applications. Another way is to quantify the minimum resources that can be allocated, this will result into transmission of small data units, thus increasing the energy efficiency. Also, energy efficiency strongly depends on the transmission power and the adaptive modulation/coding procedure. Indeed, when an aggressive modulation is used, the energy efficiency decreases for the transmission of small amounts of data, as the physical resource blocks are under utilized. Also, there will be an optimum frame length which minimizes the average delay and maximizes the energy efficiency of the M2M network.

Another challenge is of implementing and optimizing the wireless resource sharing between the M2M and H2H links. An orthogonal resource allocation is simple to implement and does not cause any interference among both types of users. However, the resource efficiency increases when using non-orthogonal sharing. Where both types of users are sharing the same frequency and time, it results in interference thus costing efficiency. The severity of interferences depends on two major factors; transmit powers of M2M transmitters and specific resource allocation protocol. In order to guarantee the priority of cellular communication, keeping the transmit powers of M2M transmitters under specific thresholds is one direct way to ease its interference to cellular receivers. However, this measure seriously harms the reliability of M2M communication. Therefore, attention must be focused on the design of resource allocation scheme, and effort should be taken to make eNB more intelligent.

Data aggregation schemes i.e., group based MAC has been suggested by many researchers and seems efficient and applicable for massive M2M access. Group leader accumulates the received uplink traffic and periodically merges the messages at its own internal buffer. It can reduce signaling overhead and simplifies the traffic, policing and charging thus increasing the energy efficiency of MTC devices. There are numbers of available algorithms on clustering already in use by WSNs, that can be equally applicable for MTC devices. However, it is

challenging to make/remake groups/clusters in real time with specific QoS requirements.

Currently extended access barring is considered as the good choice that could control the potential surge of access attempts, and therefore it may be introduced as a baseline solution to RAN overload in the future. The basic idea of EAB is that the MTC devices belonging to certain access classes indicated by the network broadcasting information are not permitted to access the network as long as EAB is activated. However, the enabling mechanism and practical barring procedures for EAB are still in development. Broadcasting MTC access barring by eNB is one of the solutions; however, this solution does not prevent congestion if the devices try to connect simultaneously when the barring period elapses.

In addition to the above, prioritized random access for pre-allocating RACH for avoiding simultaneous access attempts and the use of cooperative access class barring to offload the excessive M2M traffic are proposed. However, these schemes are infeasible for static MTC devices and do not accommodate additional traffic from H2H devices. Critical step in implementing these schemes is to estimate the number of active MTC devices or ones triggered by an event, and optimize the transmission probability in medium access. This problem is even more important in event-driven M2M applications, which are characterized by highly bursty traffic. However, these schemes work well under some conditions and seem infeasible in others. In the given situation, distributed random access techniques may be a better choice than scheduling.

Another research area is mainly on efficient congestion control and access delay during channel access attempts. Still important issues such as fairness in resource allocation, maximization of throughput, energy efficient MTC device design, etc are yet to be explored. Keeping in mind the versatile M2M application paradigm, it is very difficult to devise a single generalized solution for all quality measures.

Intelligent power control scheme is required to minimize interference from M2M to H2H layer. Power control needs to be complemented by mode selection, resource scheduling and link adaptation to properly handle intra and inter-cell interference. Building M2M power management on already standardized and widely deployed power control schemes facilitate

not only a smooth introduction of MTC devices, but would also help to develop interoperable solutions between different devices and network equipments. However, due to new interference scenarios in M2M networks, the question naturally arises whether the available power control techniques are suitable for M2M communications.

## 2.6 Chapter Summary

In this chapter, a comprehensive survey and comparison of existing techniques for emerging M2M communication is presented from the medium access control point of view. It covers majority of noble research in channel access including scheduling as well as random access techniques, and then the resource allocation for massive MTC device access. Since there are many dimensions along which to classify the channel access mechanism, we do not attempt to cover all the possible combinations. Rather, we summarize the choices made by the surveyed mechanism and tried to extract the important objectives from the literature. We tried to list M2M enabled LTE with possible emerging applications and it is an interesting research area to pursue. More efficient and less complex solutions such as combination of scheduled and random access for M2M communication over LTE could be examined towards more flexible and diverse M2M traffic and application handling.

In the following chapter, we consider MTCs in a clustered network structure, where they are divided into clusters and the devices belonging to a cluster communicate to cluster head (or controller). We propose a multi-objective optimization based solution to the problem of resource allocation in interference-limited M2M communication. We formulate the problem as a mixed-integer non-linear problem with multiple objectives and solve it using meshed adaptive direct search algorithm.

# Chapter 3

## MTCD Controller Resource Allocation

In this chapter, we propose a multi-objective optimization based solution to the problem of resource allocation in interference-limited machine to machine (M2M) communication. We consider machine type communication devices (MTCDs) in a clustered network structure, where they are divided into clusters and the devices belonging to the cluster communicate to cluster head (or controller). The cluster head aggregates the traffic and relays from MTCDs to eNB and vice versa. We maximize the number of admitted MTCD controllers and throughput with least interference caused to conventional human users. We formulate the problem as a mixed-integer non-linear problem with multiple objectives and solve it using meshed adaptive direct search algorithm. Simulation results show the effects of varying different parameters on cumulative throughput and the number of admitted MTCD controllers.

### 3.1 Introduction

The M2M application portfolio includes a vast number of diverse interconnected smart devices, each with only a small amount of data needing to be transmitted. Massive number of M2M terminals, trying random access (RA) for channel resources at once, will result in

overload, congestion and access problem. Other potential issues are on-the-air interface including physical layer transmissions, random access procedure, scheduling and radio resource allocation with quality of service support.

A lot of research has been conducted in M2M domain from the channel access, overload control and energy efficiency perspective as discussed in chapter 2. For instance, congestion and overload control was dealt with in [82, 84], while access delay was taken care in [70, 74]. One of the promising approaches for overcoming the problem of spectrum scarcity and overload problem is the grouping of MTCs into clusters. In this context, cooperative and group based access is proposed to ensure low uplink energy consumption of MTC devices in [3, 77, 78]. A similar grouping scheme is also used in [14], where joint massive access control and resource allocation schemes are proposed. In the same spirit, authors in [105–107] presented the idea of joint optimization of cluster formation and power control. In continuation to the above, a joint energy-saving mechanism, employing clustering is proposed for M2M communication in [108, 109]. In [110], power allocation and energy maximization is studied as a non-convex optimization problem in M2M networks, taking into account the circuit energy consumption and battery storage capacities.

Most of the existing M2M work has predominantly been designed with the objectives of maximizing the service quality and data rate for individual machines, minimization of machine energy consumption and maximization of the number of MTCs with minimum data rate constraint. In our work, we formulate and solve the multi-objective optimization problem of maximizing the number of admitted devices with dual interference and power constraints for human and machine devices, for M2M communications. The proposed multi-objective framework optimizes two conflicting objectives. The first objective is to maximize the total throughput and second is to minimize the total transmitted power for MTCs. In addition, we want to maximize the number of MTC controllers that can simultaneously access the channel to the base station (eNB). To best of our knowledge this type of a work is not done by others before. Previous works are comprised of optimization of channel access, resource allocation and cluster formation with a single objective, usually of energy

minimization or power control etc., but we are optimizing two important system parameters with simultaneous admission control. This type of problem is a mixed integer programming problem, which is usually NP-hard. We use Optimization Interface Toolbox (OPTI toolbox) for solving this mixed integer non-linear programming (MINLP) problem. Lot of applications of MINLP are in engineering design problems, operations research and management [111, 112]. OPTI toolbox is based on mesh adaptive direct search (MADS) [113–117] and we are using it for solving our optimization problem, which has not been used in M2M research for optimization of channel access and resource allocation.

The remainder of this chapter is organized as follows. Section II describes the system model. Then multi-objective optimization problem is formulated in Section III. Simulation setup and results are covered in Section IV. Finally, Section V provides concluding remarks.

## 3.2 System Model

We consider M2M communication supported in the traditional cellular systems. This is a high machine density scenario with hundreds of machines within a macro cell. A single eNB (base station) needs to support human devices as well as machine devices in the cell. One major problem for this scenario is the limited random access to resources and also, eNB may not be able to process massive number of access attempts generated by the machines (MTCDs). As a result, a clustered structure has been proposed to divide machines into several clusters. Note that formation of clusters on the basis of energy efficiency and QoS is another interesting area. For simplicity of analysis, here we assume that clusters are formed on the basis of MTCD spatial distribution. Within each cluster, a cluster head known as MTCD controller is elected to process the data transmission to machine devices. Therefore, the data from eNB is firstly transmitted to the MTCD controllers and then relayed to MTCDs. This results in off-loading the eNB's processing load, random access attempts and alleviate collisions. From eNB perspective, MTCD controller acts like other human devices

in the cell since they communicate directly to it through the regular resource allocation process.

We consider a heterogenous wireless network with one eNB,  $K$  MTCDS controllers and  $M$  human devices as shown in Figure 3.1. MTCDS are arranged into  $K$  clusters and each controller is responsible for the data transmission and scheduling for each device in a cluster. For simplicity of analysis, we assume that all the controllers have same data rate requirements.

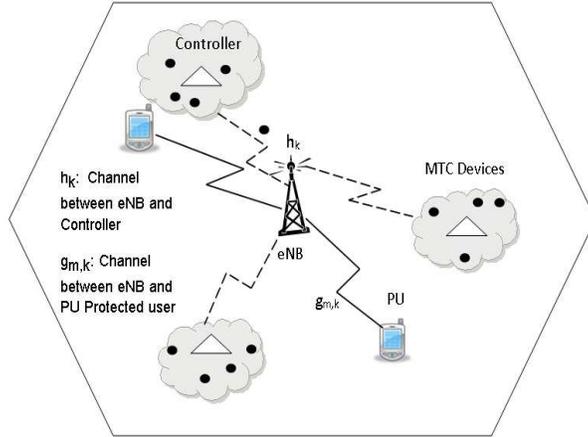


Figure 3.1: Heterogenous Network with eNB, MTCDS and Controllers

We consider the downlink transmission scenario for both human and machine links (i.e, the eNB and cluster heads act as potential transmitters) in downlink. Channels from eNB to human devices (HD) and MTCDS controllers, and controllers to MTCDS in clusters are downlinks. <sup>1</sup> We assume that MTCDS controllers are using orthogonal channel assignment from eNB, i.e., no two controllers are using the same channels. Therefore, inter-cluster interference is eliminated. We denote by  $h_k$ , the channel from the source to the  $k^{th}$  MTCDS controller,  $g_{m,k}$  the channel from the source to the  $m^{th}$  primary (human) user. Also,  $p_k$ , the source power in the  $k^{th}$  MTCDS channel and  $P_s$ , the maximum source power i.e,  $\sum_k p_k \leq P_s$ . The noise power is  $N_o$  in each channel. In Table 3.1, all the used parameters are listed.

<sup>1</sup>Note that HD can be human users or any protected users

Table 3.1: Notations

Symbol	Definition
$MTCDs$	Machine type communication devices
$MINLP$	Mixed integer non linear programming
$M2M$	Machine to machine
$eNB$	Evolved node base station
$RA$	Random access
$K$	Number of MTCD controllers
$M$	Number of human devices(HD)
$HD$	Human devices
$N_o$	Noise power
$\Phi_{total}$	Total numbers of admitted MTCDs
$I_{m,k}$	Interference threshold at $m^{th}$ HD for channel used by $k^{th}$ controller
$h_k$	Channel between the eNB and the $k^{th}$ MTCD controller
$p_k$	Maximum allowed transmission power at the $k^{th}$ MTCD channel
$P_s$	Maximum available power of the eNB
$g_{m,k}$	Channel between the eNB and the $m^{th}$ HD in the $k$ MTCD channel
$F()$	Fitness function
$C_k$	Channel capacity of the $k^{th}$ controller
$C_{th}$	Minimum capacity threshold for MTCD controllers
$C_{total}$	Total throughput available from MTCD controllers
$\beta$	Path loss component
$\psi$	Rayleigh random variable
$K_o$	Constant based on antenna characteristics
$d_m$	Distance between eNB and $m^{th}$ HD
$R$	Radius of protected area of $m^{th}$ HD, $R_m = R$

### 3.3 Problem Formulation and Solution

The problem that we are considering in this chapter is based on maximizing the number of devices in a particular M2M scenario. We also consider maximization of total throughput (sum data rate) available from the selected devices, minimization of the interference caused to the human users and minimization of the transmission power. The problem can be described as follows:

#### 3.3.1 Problem Description

1) **Maximizing the total throughput:** Consider the channel capacity of the  $k^{th}$  controller for the shared channel:

$$C_k = \log \left[ 1 + \frac{p_k}{N_o} (|h_k|^2) \right], \forall k = 1, 2, \dots, K \quad (3.1)$$

where  $h_k$  is the channel gain between the source and the  $k^{th}$  controller.

The total throughput available from MTCD channel is the sum of throughput available from each channel. i.e,

$$C_{total} = \sum_{k=1}^K C_k \quad (3.2)$$

From (3.2), we can see that  $C_k$  can be maximized by changing the maximum allowed transmission power  $p_k$ . Therefore, our first objective can be written as:

$$\max_{p_k} : C_{total} \quad (3.3)$$

Note that as all devices may not be admitted in a certain channel, and hence we introduce a variable  $x_k$  such that:

$$x_k = \begin{cases} 1 & \text{if } k^{th} \text{ controller is admitted,} \\ 0 & \text{otherwise.} \end{cases} \quad (3.4)$$

In this case (3.2) can be written as:

$$C_{total} = \sum_{k=1}^K x_k C_k \quad (3.5)$$

and (3.3) can be written as:

$$\max_{x_k, p_k} : C_{total} \quad (3.6)$$

**2) Reduction in the transmission power of each controller:** It will reduce the interference from devices in  $k^{th}$  MTCD channel to  $m^{th}$  HD.

We require that total power from all MTCD controllers should be less than the total available power of source, i.e,

$$\sum_{k=1}^K x_k p_k \leq P_s \quad (3.7)$$

Transmission power  $p_k$  should be such that it creates interference less than the interference threshold at the the  $m^{th}$  HD, i.e,

$$p_k \leq \left[ \frac{I_{mk}}{g_{m,k}} \right] \text{ for } \quad \forall k = 1, 2 \dots K, m = 1, 2 \dots M \quad (3.8)$$

where  $g_{m,k}$  is the channel between the source and the  $m^{th}$  HD in the  $k^{th}$  MTCD channel. Therefore, the second objective is written as:

$$\begin{aligned} \min_{x_k, p_k} & : P_{total} \\ \text{subject to} & \\ & p_k \leq \left[ \frac{I_{mk}}{g_{m,k}} \right] \end{aligned} \quad (3.9)$$

where

$$P_{total} = \sum_{k=1}^K x_k p_k \quad (3.10)$$

**3) Maximize the number of controllers:** Effort is made to maximize the number of controllers in each channel. The total number of controllers admitted into the network is given as:

$$\Phi_{total} = \sum_{k=1}^K x_k, \forall k = 1, 2 \dots K \quad (3.11)$$

The third objective is described as maximization of (3.12), i.e,

$$\max_{x_k} : \Phi_{total} \quad (3.12)$$

We want to combine the above mentioned objectives; however, all of these objectives have different units and variations. Therefore, it is not possible to combine them as such. We normalize each objective value between 0 and 1 such that these could be combined. The first objective is normalized by the maximum throughput available, i.e.,

$$C_{total}^{max} = \sum_{k=1}^K C_k^{max} \quad (3.13)$$

Using (3.13), we can rewrite (3.6) as:

$$F_1 = \max_{x_k, p_k} \frac{C_{total}}{C_{total}^{max}} \quad (3.14)$$

Similarly, second objective can be normalized by maximum source power  $P_s$ , consequently (5.1) can be rewritten as:

$$F_2 = \min \frac{P_{total}}{P_s}$$

subject to

$$p_k \leq \left[ \frac{I_{mk}}{g_{m,k}} \right] \text{ for } \quad \forall k = 1, 2 \dots K, m = 1, 2 \dots M \quad (3.15)$$

The third objective is normalized by total number of MTCD controllers and is rewritten as:

$$F_3 = \max_{x_k} \frac{\phi_k}{K} \quad (3.16)$$

To make (3.16) into a joint minimization problem, we convert  $F_1, F_3$  into a minimization problem by rewriting them as  $(1 - F_1)$  and  $(1 - F_3)$ . We further introduce weights for each objective which are based on the emphasis we want to place on each objective. In this case, joint objective function is given as:

$$F = w_1 (1 - F_1) + w_2 F_2 + w_3 (1 - F_3), \quad (3.17)$$

where  $w_1, w_2$  and  $w_3$  are input weights for each objective function and:

$$w_1 + w_2 + w_3 = 1 \quad (3.18)$$

Making use of (3.17), the joint multiobjective optimization problem can be rewritten as:

$$\begin{aligned}
& OP1 : \\
& \min_{x_k, p_k} \quad F \\
& \text{subject to} \\
& C1 : p_k \leq x_k P_s, \forall k \\
& C2 : \sum_{k=1}^K p_k \leq P_s \\
& C3 : p_k |g_{m,k}|^2 \leq I_{m,k}, \forall (m, k) \\
& C4 : C_k \geq x_k C_{th}, \forall k \\
& C5 : x_k \in [0, 1],
\end{aligned} \tag{3.19}$$

where we have added additional constraints C1, C2 and C4. The constraints can be described as:

- C1 ensures that the power in each selected channel is less than the maximum available source power.
- C2 ensures that the total transmission power in all the channels is less than the maximum source power. Note that C1 ensures that any non-selected MTCD controller will have zero power. In the absence of C1, non-selected controllers can still have non-zero transmission power.
- C3 ensures that interference caused by devices in  $k^{th}$  channel to the  $m^{th}$  HD is less than a given interference threshold.
- C4 ensures that the throughput in each channel is greater than a minimum threshold  $C^{th}$ .
- C5 ensures that  $x_k$  is a binary integer variable.

The optimization problem in OP1 is a class of mixed integer programming problem [118] that is generally NP-hard. Once discrete realization is known, the rest of the continuous

optimization problem is a convex optimization problem [119]. The optimal solution for discrete variables of the above problem can be computed using branch and bound method [120] or exhaustive search but their computational complexity is very high. Also for every discrete realization of OP1, we have to solve a constraint continuous optimization problem. A number of algorithms are available to solve mixed integer programming problems with box constraint only. In our work, we use meshed adaptive direct search (MADS) algorithm to solve OP1.

### 3.3.2 Proposed Solution

Meshed adaptive direct search (MADS) is iterative derivative-free global optimization algorithm that uses direct method with polling and searching criterion. Each iteration of MADS evaluates a finite set of mesh points in the solution search space. A polling step is used after mesh search that leads to convergence of this solution. Let us first describe the MADS notations used in this chapter.

- 
1. ITER: Maximum number of iterations and  $i$  is the  $i$ th iteration.
  2.  $M_i$ : Set of mesh points at the  $i$ th iteration.
  3.  $\Gamma_i$ : Set of poll points at the  $i$ th iteration.
  4.  $\Delta_i^m \in \mathcal{R}^+$ : Mesh size parameter at the  $i$ th iteration
  5.  $\Delta_i^p \in \mathcal{R}^+$ : Poll size parameter at the  $i$ th iteration
  6.  $\Theta^m$ : Mesh contraction/expansion factor
  7.  $\Theta^p$ : Poll contraction/expansion factor
  8.  $D_i^m$ : Finite set mesh direction matrix at the  $i$ th iteration
  9.  $D_i^p$ : Finite set poll direction matrix at the  $i$ th iteration

---

**Algorithm 1 : MADS**

---

1: **Initialization:**

2:  $i \leftarrow 0, \Delta_i^m > \Delta_i^p > 0$

3:  $Terminate \leftarrow FALSE, ImprovedFound \leftarrow FALSE$

4: Initialize  $\mathcal{Z}_i^m, \mathcal{Z}_i^p, 0 < \Theta^m < \Theta^p < 1$

5:  $F \leftarrow w_1(1 - F_1) + w_2F_2 + w_3(1 - F_3)$

6: **Algorithm Execution:**

7: **while**  $Terminate == FALSE$  **do**

8:  $M_i = \bigcup_{z \in \beta_i} \{z + \Delta_i^m D_i^m\}$

9:  $\mathcal{Z}_i^m \leftarrow \begin{cases} \arg \min_{z \in M_i} F \\ \text{subject to C1, C2, C3 and C4 of (3.19);} \end{cases}$

10: **if**  $F(\mathcal{Z}_i^m) > F(\mathcal{Z}_{i-1}^m)$  **then**

11:  $\Gamma_i = \bigcup_{z \in \beta_i} \{z + \Delta_i^p D_i^p\}$

12:  $\mathcal{Z}_i^p \leftarrow \begin{cases} \arg \min_{z \in \Gamma_i} F \\ \text{subject to C1, C2, C3 and C4 of (3.19);} \end{cases}$

13: **if**  $F(\mathcal{Z}_i^p) < F(\mathcal{Z}_{i-1}^p)$  **then**

14:  $ImprovedFound \leftarrow TRUE$

15: **end if**

16: **else**

17:  $ImprovedFound \leftarrow TRUE$

18: **end if**

19: **if**  $ImprovedFound == TRUE$  **then**

20:  $\Delta_{i+1}^m \leftarrow \Delta_i^m / \Theta^m, \Delta_{i+1}^p \leftarrow \Delta_i^p / \Theta^p$

21: **else**

22:  $\Delta_{i+1}^m \leftarrow \Delta_i^m \Theta^m, \Delta_{i+1}^p \leftarrow \Delta_i^p \Theta^p$

23: **end if**

24:  $ImprovedFound \leftarrow FALSE$

25:  $i \leftarrow i + 1$

26: **if** Termination Criterion Satisfied **then**

27:  $Terminate \leftarrow TRUE$

28: **end if**

10.  $\beta_i$ : Set of finite points at the  $i$ th iteration to evaluate the optimization objective
11.  $z$ : Any arbitrary point on the solution space
12.  $n$ : The number of variables

The MADS algorithm is a kind of iterative contraction and expansion algorithm. The algorithm searches the solution in different directions on predefined neighborhood. By getting better solution in the neighbor, MADS contracts its searching region across the better solution point, otherwise it expands the neighbor search space. A detailed description of the algorithm is given in Algorithm 1. As mentioned above that to get optimal solution, either we need branch and bound method or exhaustive search algorithm for discrete variables. The complexity of worst case branch and bound algorithm is same as exhaustive search which increases exponentially with the number of discrete variables. In case of optimization problem in OP1, their complexity is  $O(2^K)$ . It is shown in the literature that MADS converges to  $\varepsilon$ -optimal solution in finite steps [121] [122]. In [123], based on Clarke's [124] calculus for non-smooth optimization problem, authors prove that MADS converges to global optimum with error tolerance of  $\varepsilon$ . They also show that the convergence of MADS is independent of start point. The complexity of MADS to get  $\varepsilon$ -optimal solution is  $O(\frac{K^2}{\varepsilon})$  [121]. This computational complexity is much lower than the complexity of branch and bound method or exhaustive search algorithm.

## 3.4 Results and Discussion

### 3.4.1 Simulation Setup

Figure 3.2 shows the simulation setup layout with one eNB in the centre shown by a triangle. Two human users (HDs) shown by squares and multiple MTCN controllers shown by stars are uniformly distributed in the entire area. In all the simulations, the channels between source and destination have independent distribution. Some common parameter values for

simulation are shown in Table 3.2. Channel gain  $h$  is modeled as:

$$h = \Psi K_o \left( \frac{d_o}{d} \right)^\beta, \quad (3.20)$$

where  $K_o$  is a constant that depends on the antenna characteristic and average channel attenuation,  $d_o$  is the reference distance for the antenna far field,  $d$  is the distance between transmitter and receiver,  $\beta$  is the path loss constant and  $\Psi$  is the Rayleigh random variable. Since this formula is not valid in the near field, in all the simulation results, we assume that  $d$  is greater than  $d_o$ . In all the results,  $d_o = 10\text{m}$ ,  $K_o = 50$  and  $\beta = 2$ . The HD's protected distance  $R_m$  is set to 10m for all  $M$ . MTCD controllers and HDs are uniformly distributed. For each HD, there is a HD protection area wherein the strengths of MTCD controllers must be constrained. We define as  $R$ , the radius of the protected circular area for each individual HD. Given a distance  $d_m$  between the eNB and the  $m^{\text{th}}$  HD and the radius  $R_m$  of the protected circular area of the  $m^{\text{th}}$  HD, the channel from the source to  $m^{\text{th}}$  HD in  $k^{\text{th}}$  channel is given as:

$$g_{m,k} = \frac{\tilde{g}_{m,k}}{(d_m - R_m)^\beta}, \quad (3.21)$$

where  $\tilde{g}_{m,k}$  is the small scale fading and  $\beta$  is the path loss exponent .

Table 3.2: Common Parameter Values

Parameters	Values
$P_s$	10W
$I_{m,k}$	{10mW, 100mW}
$K$	2,4,6,8,10
$M$	1, 3
$C_{th}$	5
$N_o$	$10^{-6}\text{W}$
Bandwidth	1MHz

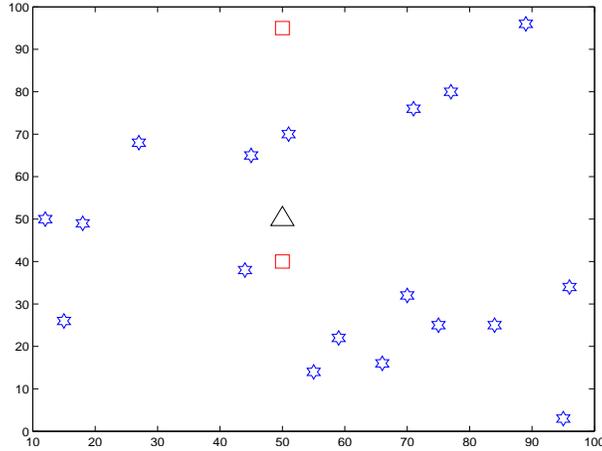


Figure 3.2: Layout for M=2

### 3.4.2 Results and Discussion

We first show the effect of weights on sum rate, sum power and number of admitted MTCs. We vary  $w_1$ ,  $w_2$  and  $w_3$  and solve the optimization problem corresponding to these weights. We simulate for four different sets of weights:  $w_1 = 0.1$ ,  $w_2 = 0.3$  and  $w_3 = 0.6$ ,  $w_1 = 0.33$ ,  $w_2 = 0.33$  and  $w_3 = 0.3$ ,  $w_1 = 0.5$ ,  $w_2 = 0.25$  and  $w_3 = 0.25$ , and  $w_1 = 0.8$ ,  $w_2 = 0.1$  and  $w_3 = 0.1$ . These sets can be used to demonstrate the effects on the optimization results by varying the weights.

We present sum rate and the number of admitted MTCs versus total number of MTCs in Figure 3.3. The total number of MTCs is varied from two to ten in a step of two. The optimization problem (OP1) is solved with different sets of weights described above and the obtained values of  $x_k$  and  $p_k$  for all values of  $k$  are used to calculate the sum rate. From Figures 3.3(a)-3.3(c), the following observations can be made:

- It can be seen in Figure 3.3(a) that when  $w_1$  is less than both  $w_2$  and  $w_3$ , the sum rate is the lowest.
- When  $w_1$  and  $w_2$  are equal and both are greater than  $w_3$ , the sum rate increases in Figure 3.3(b) compared to Figure 3.3(a).

- The increasing trend continues as  $w_1$  becomes greater than both  $w_2$  and  $w_3$ , and is maximum for  $w_1 = 0.8$  in Figure 3.3(d) as in this case a lot of emphasis is placed on maximizing the sum rate.
- It can be further seen that the sum rate is decreasing as the number of MTCDS increases. This is especially evident in Figure 3.3(a). The reason is that as  $w_3$  is the largest in this case, most of the emphasis is placed on maximizing the number of admitted MTCDS, followed by the minimization of power and maximization of capacity. Since more emphasis is placed on minimization of power and as the number of admitted users increases, the power  $p_k$  of each controller will be less with increasing number of users. As a result, the sum capacity decreases with increasing number of users in Figure 3.3(a).
- In Figures 3.3(b) and 3.3(c), a similar trend as in Figure 3.3(a) can be noted although the decrease in sum rate with respect to the maximum sum rate is less. In Figure 3.3(c), for a number of admitted MTCDS equal to ten, the sum rate increases. In Figure 3.3(d), as  $w_1$  is much higher than  $w_2$  and  $w_3$ , more emphasis is placed on sum rate maximization and as a result, an increasing trend for the sum rate can be seen.
- A linear trend with respect to the number of admitted MTCDS can be seen as all the MTCDS are admitted.

Next we present a comparison of sum rate and sum power for the same sets of weights as used for Figure 3.3. The results are shown in Figure 3.4 and the following observations can be made:

- It can be seen that when  $w_2$  is greater than or the same as  $w_1$ , the sum power is almost the same with increasing number of MTCDS. Normally, this power should increase as more users will consume more power. However, the higher emphasis placed on  $w_2$  stops this increase from happening.

- As  $w_2$  decreases, the sum power increases in Figures 3.4(c)-3.4(d). In Figure 3.4(d), the sum power is the highest as little emphasis is placed on minimizing it and more emphasis is placed on maximizing sum rate. The increase in power may be offset by a much higher increase in sum rate.
- It can be seen in Figure 3.4(c) that the sum power increases for a number of MTCDs=10, which explains the reason why sum rate increases in Figure 3.3(c).

We present the number of admitted users for different rates and interference thresholds for  $w_1 = 0.8$ ,  $w_2 = 0.1$ ,  $w_3 = 0.1$ , in Figure 3.5. Figure 3.5(a) shows the results for an interference threshold of 10 mW. For a rate of 250 kbps, the number of admitted users is the same as the number of MTCDs, hence a linear relationship can be seen. For the rate of 500 kbps, this relationship is not linear and the number of admitted users is less than the number of MTCDs, especially for the number of MTCDs equal to 6 and 10, the number of admitted MTCDs is 5 and 8, respectively. In Figure 3.5(b), the same results are presented for an interference threshold of 100 mW. It can be seen again that for a rate of 250 kbps, a linear relationship can be seen, whereas for a rate of 500 kbps, this relationship is non-linear as the number of admitted MTCDs is less than the number of MTCDs for 6, 8 and 10 MTCDs. However, the number of admitted MTCDs is higher for 10 MTCDs for the higher interference threshold. We can make the following observations:

- A higher rate leads to a lower number of admitted MTCDs as higher rate requires higher power. However, due to interference constraints, the power is limited and hence a lower number of users are admitted to satisfy the constraint.
- A lower interference threshold causes the constraint  $C2$  to be more strict, resulting in less number of admitted MTCDs.

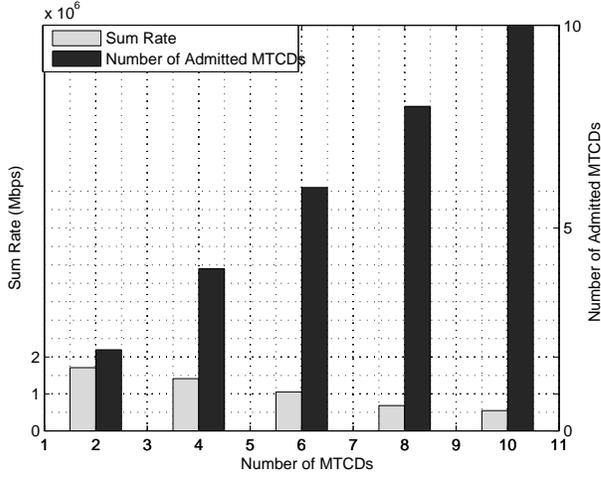
Figure 3.6 shows the number of admitted MTCDs versus the number of MTCDs for two cases: number of HDs equal to one and number of HDs equal to three. For  $M=1$ , i.e., a single HD, the number of admitted MTCDs is the same as the number of MTCDs, whereas

for  $M=3$ , the number of admitted MTCs is less than the number of MTCs for 8 and 10 MTCs. Hence, the relationship between number of admitted MTCs and the number of HDs is inversely proportional. The reason for this is that a higher number of HDs may lead to more constraints due to higher number of channels. As a result, some of the channels may be worse than others leading to stricter constraints, resulting in less number of MTCs being admitted.

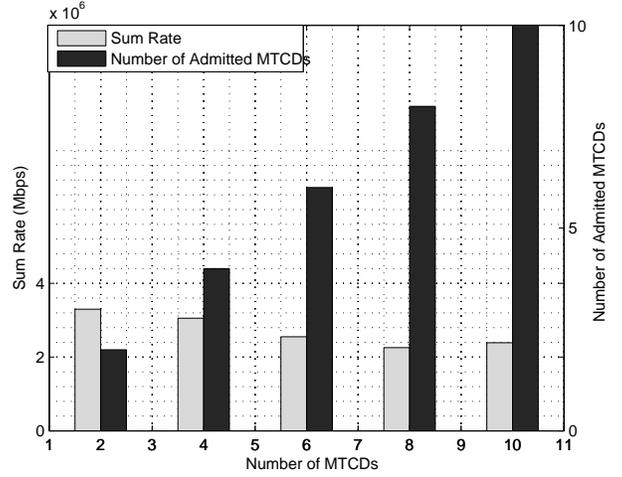
### 3.5 Chapter Summary

In this chapter, we considered a clustered network consisting of H2H (HDs) and machine type communication devices (MTCs), where all MTCs communicate with a cluster head. Cluster head is responsible to receive the aggregated data from eNB and relay them to its associated devices. We further formulated the problem of resource allocation in interference-limited M2M communication as a mixed-integer non-linear problem having multiple objectives. The multiple objectives were combined using weights, where these objectives are to maximize the cumulative throughput, reduce the transmission power for minimizing interference to HDs and maximize the number of controllers. We solved the problem using mesh adaptive direct search algorithm. Simulation results showed the dependence of cumulative throughput and the number of admitted devices with respect to different weights and other factors, such as interference threshold and different number of HDs.

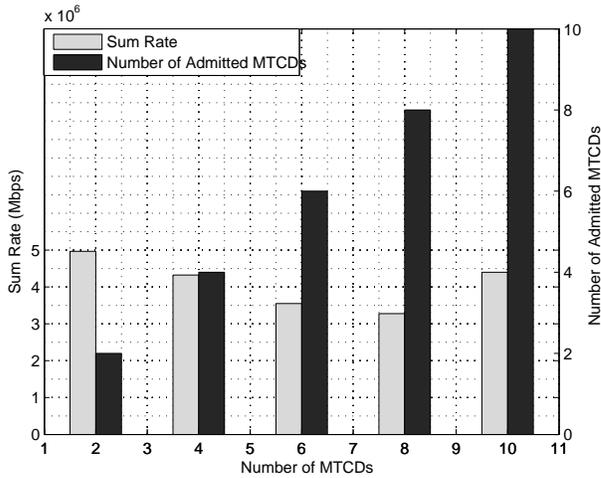
In the following chapter, we apply a Q-learning algorithm to carry out slot assignment for MTCs in M2M communication. We first make use of a K-means clustering algorithm to overcome the congestion problem in an M2M network where each MTC is associated to one controller. Subsequently we formulate the slot selection problem as an optimization problem. Then we present a solution using the Q-learning algorithm to select conflict-free slot assignment in a random access network with MTC controllers.



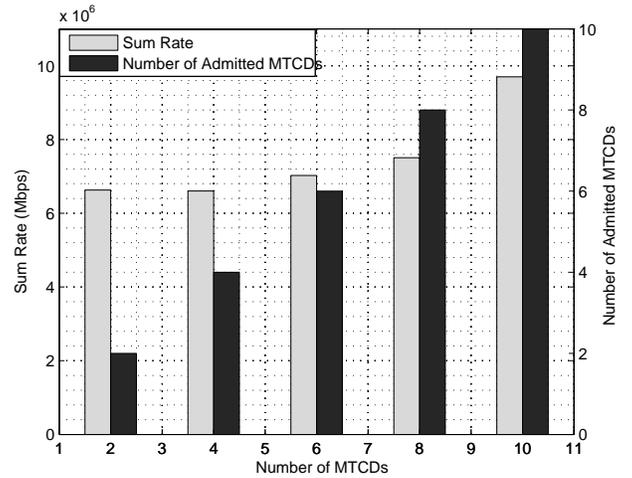
(a)



(b)

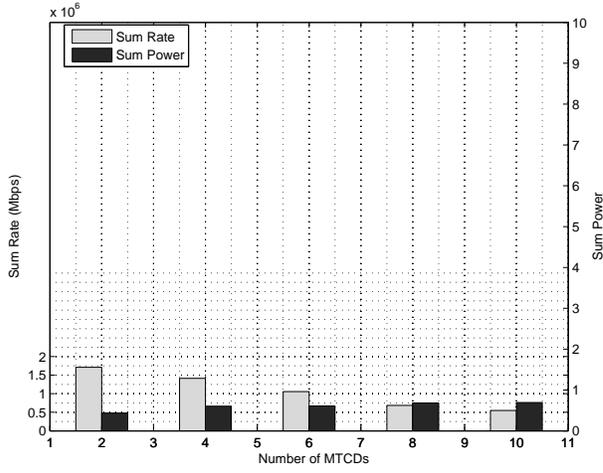


(c)

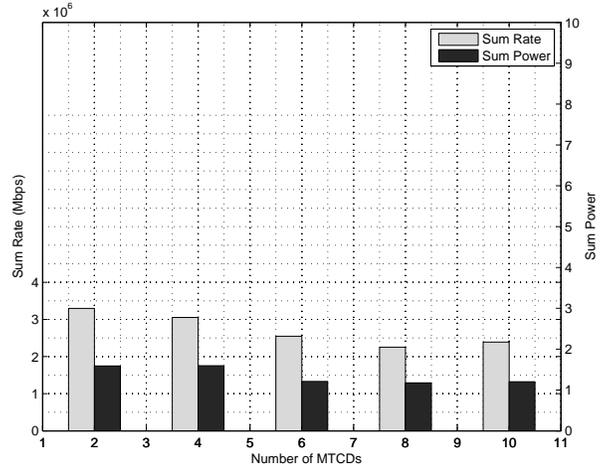


(d)

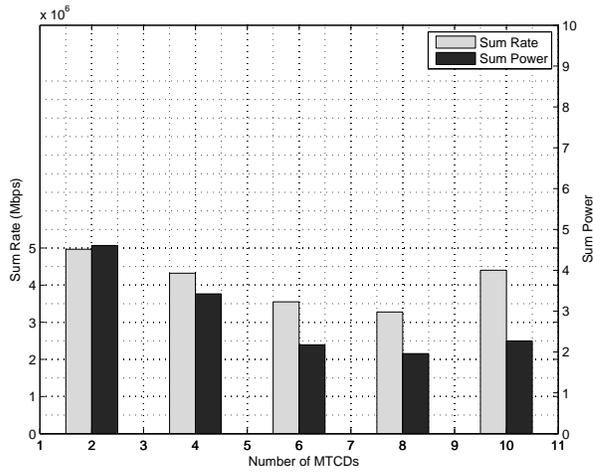
Figure 3.3: Sum Rate and Admitted MTCD for different set of weights: Number of HDs = 1, With the increase in the  $w_1$ , the sum rate will increase. (a)  $[w_1 w_2 w_3] = [0.1 0.3 0.6]$ , (b)  $[w_1 w_2 w_3] = [0.33 0.33 0.3]$ , (c)  $[w_1 w_2 w_3] = [0.5 0.25 0.25]$ , (d)  $[w_1 w_2 w_3] = [0.8 0.1 0.1]$



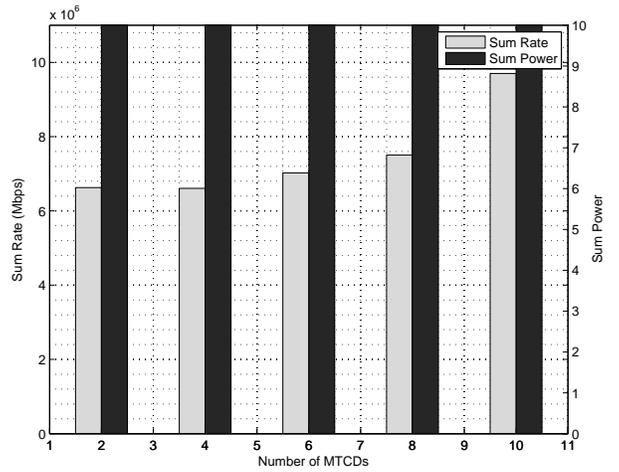
(a)



(b)

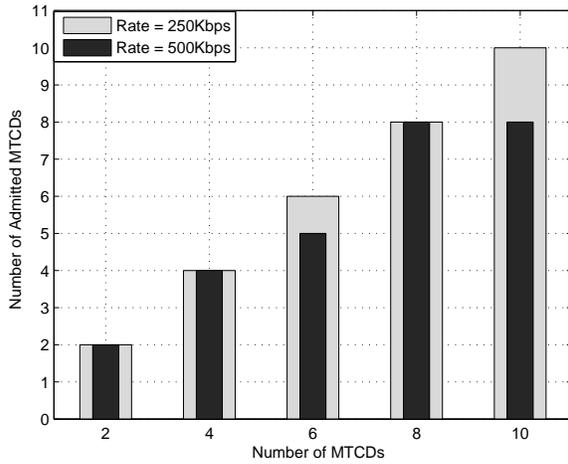


(c)

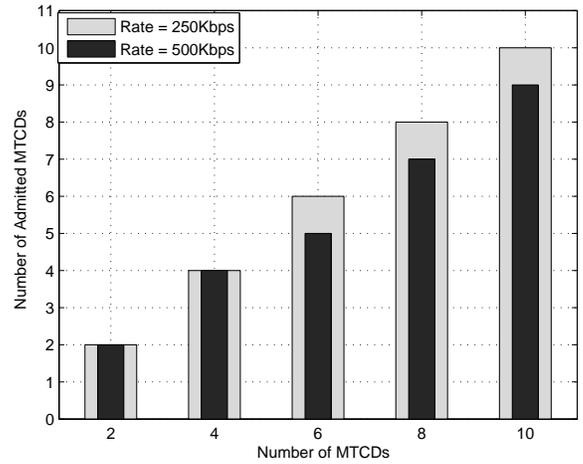


(d)

Figure 3.4: Sum Power and Sum Rate for different set of weights : Number of HDs = 1, Less value of  $W_2$  means less importance to power optimization. (a)  $[w_1 w_2 w_3] = [0.1 0.3 0.6]$ , (b)  $[w_1 w_2 w_3] = [0.33 0.33 0.3]$ , (c)  $[w_1 w_2 w_3] = [0.5 0.25 0.25]$ , (d)  $[w_1 w_2 w_3] = [0.8 0.1 0.1]$



(a)



(b)

Figure 3.5: Admitted MTCD for different rate and interference. (a)  $[w_1 \ w_2 \ w_3] = [0.8 \ 0.1 \ 0.1]$ , Interference = 10mW (b)  $[w_1 \ w_2 \ w_3] = [0.8 \ 0.1 \ 0.1]$ , Interference = 100mW: Less interference threshold means less number of admitted MTCDs

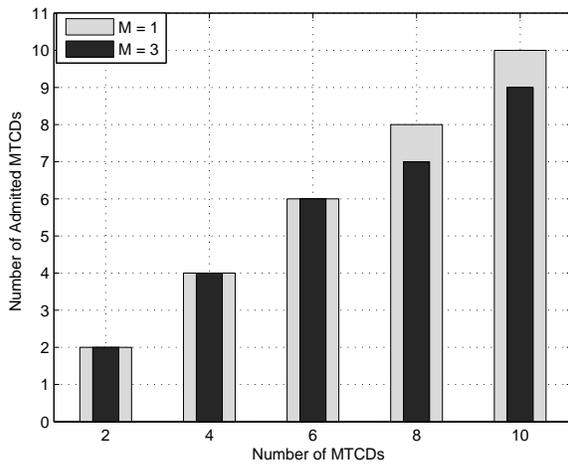


Figure 3.6: Admitted MTCD for different number of human users: Less HDs means more number of admitted MTCDs  $[w_1 \ w_2 \ w_3] = [0.33 \ 0.33 \ 0.33]$ , Interference = 10mW

# Chapter 4

## Slot Allocation without Interference

In the last chapter, a multi-objective optimization for resource allocation in M2M networks was discussed. One such objective is the number of admitted users (MTCDs) in the M2M network under a cluster. We now assume that proper number of MTCDs have been admitted and then the problem of time slot allocation among them is then addressed without any inter-cluster interference.

In the considered M2M system, we apply a reinforcement learning algorithm to carry out slot assignment for machine type communication devices. We first make use of a K-means clustering algorithm to overcome the congestion problem in an M2M network where each MTCD is associated to one controller. Subsequently, we formulate the slot selection problem as an optimization problem. Then we present a solution using the Q-learning algorithm to select conflict-free slot assignment in a random access network with MTCD controllers. The performance of the solution is dependent on parameters such as learning rate and reward. We analyze the performance of the proposed algorithm considering different parameters related to its operation. The convergence time, i.e., the time required to reach a solution decreases with increasing value of learning rate, whereas the convergence probability increases. In addition, for smaller values of learning rate, the convergence time decreases with increasing reward values. We also compare with simple ALOHA and channel-based scheduled allocation and show that the proposed Q-learning based technique has a higher probability of assigning

slots compared to these techniques.

## 4.1 Introduction

Massive access from large number of MTCs and resulting overhead are major concerns in future M2M networks [4]. Therefore, data aggregation becomes an important technique to achieve efficient data transmission [125]. Several researchers have employed data aggregation after forming virtual and physical clusters. Clustering is a energy efficient technique that alleviates network congestion [126–128]. An M2M network usually consists of a large number of typically power constrained devices, which gather and relay data without human interaction. The K-means clustering algorithm is one such algorithm, in which cost function usually measures the sum of the distances (squared differences) between data items and estimated centroid of clusters [46, 129, 130]. According to the initial clustering center, it classifies the objects of data set and recalculates the clustering center and the data object classification. The end of the iteration marks that the clustering criterion function has converged [131]. In [33], managing radio resources over clusters instead of individual MTCs and the corresponding benefits are presented.

In [132], persistent resource-allocation algorithm is proposed for M2M communication, which allocates resource to machines in an recursive manner based on the tree. This method reserves radio resources without any additional control signalling but it is only beneficial for periodic traffic and is not useful for aperiodic or bursty data. Most existing resource allocation methods in M2M communications are based on random competition [33, 52, 133–136]. These methods require MTCs to accomplish the random access procedures before actual resource allocation.

Q-learning algorithm is a well known reinforcement learning (RL) technique, which enables an agent (e.g., a sensor node) in M2M networks to learn by interacting with its environment. The agent will learn to take the best actions that maximize its long-term rewards by using its own experience [11]. In [54], RL-based base station selection algorithm is proposed

that allows the MTCs to choose the base station in a self-organizing fashion. While authors in [137] used QoS performance measure to switch from one base station to another and it is the ratio between the device throughput and its delay. In [138], a distributed algorithm in which MTCs share resources with a particular cellular user in a TDMA manner is proposed. Fast adaptive slotted Aloha (FASA) is proposed in [139] taking into account the knowledge of the idle/successful/collided state of the previous slots. Slotted Aloha exploiting successive interference cancellation (SIC), called Frameless Aloha, is presented in [140]. These techniques can ideally provide high performance in a M2M scenario in terms of throughput. Nonetheless, efficiency and complexity aspects are not considered. In particular, the SIC mechanism sets quite high requirements to the eNB in terms of storage and processing capabilities. In addition, this is not energy efficient as for each frame, the devices must transmit a large set of replicas of the same packet to the eNB making it inefficient.

Q-learning RACH access scheme (QL-RACH) is proposed in [141, 142] to control M2M traffic in order to reduce its impact on a cellular network. It uses Aloha and an intelligent slot assignment strategy to avoid collisions amongst the MTCs. However, the performance of this scheme is dominated by the uncontrolled H2H traffic especially at the upper load limit. Also, there is a restriction of using a virtual M2M frame having length (in time slots) equal to the number of MTCs in the network. This imposes the upper bound on the number of MTCs being served simultaneously. In addition to this, every node keeps a Q-value for each slot in the M2M frame to record the transmission history on that slot in consecutive frames, which is not an efficient mechanism for the energy constrained, battery limited nodes.

In contrast to these references, in our work, Q-learning is used for resource allocation of MTCs. Dedicated orthogonal resources are assigned to H2H and M2M users and there is no human-MTCD traffic interaction. Controllers make use of Q-learning to transmit data to their associated devices by picking up the best slot. Also, controllers are keeping record of Q-values of each slot instead of each MTCD. Slot allocation is decentralized and there is no communication involved among controllers and eNB and between controllers and

MTCDs. Only control channel ACK, NACK is involved to report collision to the controllers. In addition to this, our method does the channel access and resource allocation at the same time. It is similar to Aloha but keeps knowledge of previous success/failure. Therefore, it is more efficient in terms of admittance rate and success probability.

The motivation of this work is to develop a distributed conflict-free slot allocation scheme for MTCDs in a clustered random access M2M network and to show the reinforcement learning mechanism can be used with robust performance. Hence, we consider a clustered network in which the cluster head/controller is responsible for all communications between eNB and MTCDs, thus reducing signaling overhead. We assume that human users and machine devices are separated by orthogonal division, therefore, M2M communication is in dedicated channel thus not affecting H2H communication [61]. MTCd controllers are separated in frequency but share the same time slot for communication. There exist simultaneous communication between cluster heads and MTCDs, and between cluster heads and eNB using orthogonal resource allocation. Here, we focus on communication from cluster heads to its respective MTCDs, and study how cluster heads distributively make decisions about resource (slot) allocation. These decisions strongly depend on slot selection made by neighboring clusters as the system performance is limited by the collision caused due to simultaneous data transmission in the same frequency band and in the same time slot by all the cluster heads/controllers. This scenario can easily be mapped onto a multi-agent system [143] since it consists of multiple intelligent and autonomous agents, i.e., the cluster controllers and independent MTCDs. These agents have incomplete information and capabilities for solving the simultaneous transmission and resulting collision on their own. Also, there is no central entity in charge of providing information and global control of interference. The main contributions of this chapter are:

- We propose a self organized slot allocation based on the paradigm of independent learning, where agents are unaware of the other agents' actions in a clustered M2M network.

- Our work in comparison with the existing literature utilizes Q-learning for the resource allocation of MTCs. The advantage is the performance enhancement of MTCs through selecting the conflict-free time slots in congested scenarios using Q-learning distributively.

The rest of this chapter is organized as follows: In the next section, we describe the system model and cluster formation. Section III introduces reinforcement learning and the proposed solution. In section IV, we present simulation results convergence capabilities of the proposed approach with respect to different parameters. The convergence capabilities are defined as convergence time, convergence probability and cumulative success rate. Then section V concludes this chapter.

## 4.2 System Model

We consider an M2M network having dedicated frequency channels, separated from H2H communication. All the MTCs are arranged in form of clusters and formation of clusters is done using the K-means algorithm. A list of notations used in this paper is given in Table 4.1. Note that we use cluster head and controller interchangeably in this paper.

### 4.2.1 Clustered M2M Network

A clustered M2M network is considered in which the number of MTCs per unit area is large and a single eNB is not able to serve all the devices in an efficient manner. Therefore, all the MTCs are divided into clusters and each cluster has a cluster head/controller. Figure 4.4 shows the simple clustered M2M network as an example, in which four clusters with three devices per cluster are considered for analysis. The K-means algorithm is used for the formation of clusters because of its simplicity. A single eNB is present, which is in communication with the cluster heads/controllers. MTC controllers are responsible for data aggregation and data transmission to/from their respective MTCs.

Table 4.1: Notations

Symbol	Definition
$K$	Number of clusters/MTCN controllers
$u$	Number of devices in each cluster
$T$	Number of slots in each frame
$\alpha$	Learning rate
$D$	Total number of MTCNs in a system
$c_j$	Centroid of $j^{th}$ cluster
$U_j$	Set of MTCNs in $j^{th}$ cluster
$x_{ij}$	Binary variable to show the association of $i^{th}$ MTCN with $j^{th}$ cluster
$j$	Index of clusters
$\beta_{ij}$	Decision variable
$i$	Index of MTCNs
$C_1, C_2 \dots C_K$	Centers of clusters 1 to $K$
$w$	Number of frequencies in a cluster
$A$	Action performed by controller
$Q$	Quality of any action
$R$	Reward function

### 4.2.2 Slot/Channel Allocation

There are two layers of communication taking place, one between eNB and cluster heads and the other between cluster heads and associated devices. In this paper, we are looking into communication from cluster heads to their associated MTCs. We consider a frame having  $T$  slots and each controller will send data, received from eNB, to its associated MTCs in downlink communication. Let  $K$  be the total number of clusters,  $D$  be the total number of devices in a network and  $u$  be the maximum number of devices in each cluster. Each controller transmits data using TDMA to its associated devices. TDMA allows several devices to share the same frequency band by dividing the signal into different time slots. For simplicity of analysis we assume<sup>1</sup>  $T = K$ , i.e., the number of cluster heads is the same as the number of slots available in a frame. In our case it allows multiple controllers to share the same wireless channel used by multiple devices, by transmitting one after the other, using different time slots.

We also use multiple frequencies within the clusters when the number of MTCs is large. This will improve the system performance as a large number of devices simultaneously transmit and receive data to or from cluster heads. In case of only one frequency channel per cluster, controllers have to multiplex data for all the devices and transmit. Typically, MTCs have a large amount of data to be sent and received at one instant; if this is the case, then multiplexing will not be a feasible choice. The second method is to transmit sequentially or randomly. It will not be efficient as it will be slower for sequential transmission and needs a central entity for sequencing. Random transmission is not energy efficient in an energy constrained M2M network, as all the devices have to be active and receptive all the time. The use of multiple frequency channels within the cluster not only offloads burden from controllers by dividing all the devices into different frequency groups, but is also energy efficient.

---

<sup>1</sup>If  $K > T$ , then admission control is required to admit a certain number of controllers into the system. If  $K < T$ , then a frame with empty slots is transmitted.

### 4.2.3 Formation of Clusters

Let there be  $i = \{1, 2, 3 \dots D\}$  MTCDs forming  $j = \{1, 2, \dots K\}$  clusters. Each MTCD will associate itself to only one cluster. We denote  $i^{\text{th}}$  MTCD associated with  $j^{\text{th}}$  cluster by  $x_{ij}$ , where  $x_{ij}$  is a binary variable  $\forall_i$ , such that

$$x_{ij} = \begin{cases} 1 & \text{if } i^{\text{th}} \text{ MTCD} \in j^{\text{th}} \text{ cluster,} \\ 0 & \text{otherwise.} \end{cases} \quad (4.1)$$

$$\sum_{j=1}^K x_{ij} \leq 1, \forall_i \quad (4.2)$$

It means that each MTCD can belong to only one cluster at one time.

We use an iterative K-means algorithm for the formation of clusters. Let tuples  $N$  and  $C$  be the vector representing the positions of  $i^{\text{th}}$  MTCD and  $j^{\text{th}}$  controller, respectively. For initialization, centres  $(C_1, C_2, \dots C_K)$  are decided at random. Association of each MTCD to the  $j^{\text{th}}$  cluster is made on the basis of some decision parameter  $\beta_{ij}$ . This parameter can be shortest distance, channel gain, or signal to noise ratio (SNR).

In this paper, Euclidean distance between MTCDs and controllers is used for the formation of clusters for simplicity and any other parameters can be considered to form the clusters. Therefore, the distance between two tuples  $(N_i)$  and  $(C_j)$  is given by (4.3), where,  $n_i = (n_{xi}, n_{yi})$  and  $C_j = (c_{xj}, c_{yj})$ . This means each MTCD is associated to the nearest cluster head.

$$d_{ij} = \sqrt{(n_{xi} - c_{xj})^2 + (n_{yi} - c_{yj})^2}, \forall_i, j \quad (4.3)$$

After the first association of MTCDs with cluster heads, a new location of cluster heads is found by iterative calculation of new centroids. New centroids are formed as:

$$c_j = \frac{\sum_{i \in U_j} x_{ij} N_i}{\|U_j\|}, \forall_{i,j} \quad (4.4)$$

Splitting the equation for the  $x$  and  $y$  coordinates of the new centroid:

$$c_{xj} = \frac{\sum_{i \in U_j} x_{ij} n_{xi}}{\|U_j\|}, \forall_{i,j} \quad (4.5)$$

$$c_{yj} = \frac{\sum_{i \in U_j} x_{ij} n_{yi}}{\|U_j\|}, \forall i, j \quad (4.6)$$

where  $U_j$  is the set of MTCDs in the  $j^{th}$  cluster. The above procedure is repeated till the position of cluster heads remains the same.  $c_{xj}, c_{yj}$  and  $n_{xj}, n_{yj}$  are the  $x, y$  coordinates of the points  $C$  and  $N$ , respectively. Table 4.2 shows various steps for the K-means algorithm used.

Table 4.2: K-Means Algorithm

<b>K-means Steps</b>
1) Select initial cluster centroids (position of MTCD controllers) by choosing K data points at random.
2) Compute the distance from each MTCD to each of the K centroids using Equation (4.3).
3) Associate the MTCDs to specific cluster head by their closest distance.
4) Compute the new cluster centroids using the shortest distance of MTCDs with respective centroid.
5) Repeat until no change in the centroids of the clusters.

In this section, we considered a clustered M2M network. An algorithm for the formation of clusters is given. MTCDs in each cluster are associated with a controller. In the next section, we shall explain slot allocation among different controllers and then formulate the problem.

### 4.3 Problem Formulation

After the formation of clusters, resource allocation for MTCDs within each cluster is done. As each controller is responsible for data transmission to its associated devices, it needs a unique time slot for each transmission. As controllers and associated MTCDs are not scheduled for

data transmission/reception, a careful distributed selection of wireless channels is required to avoid collision. MTCs are separated by various frequency channels within the cluster, and controllers are separated by time slots in a frame as discussed earlier. The actual task is that controllers need to select unique time slots for data transmission to their MTCs avoiding any inter-cluster and intra-cluster collision. In the following we will discuss the slot selection by the controllers employing Q-learning technique. We consider two cases: In the first case, devices use the same frequency, while in the second case devices use multiple frequencies.

### 4.3.1 Single Frequency Case

We consider a scenario in which some selected devices are transmitting using the same frequency. For example, consider three clusters and three slots. We want to assign three devices, one from each cluster to one time slot in such a way that no two clusters assign the same slots to their devices. Consider the frame with three slots in a three-cluster M2M network as follows; where a device  $i$  in  $j^{th}$  cluster is denoted as  $U_i^j$  as shown in Figure 4.1. In this case, MTC 3 from cluster 1 is in slot 1 of the frame, while MTCs 1 and 2 of clusters 2 and 3 are in slots 2 and 3, respectively. As each slot is assigned to only one device, we consider the assignment of MTCs to be successful. Now consider the following case: In this case, as slot 1 is assigned to MTCs 3 and 1 from cluster 1 and 2, respectively, transmission is considered unsuccessful, as shown in Figure 4.2. There will be a collision between transmissions to MTCs 3 and 1 from both cluster heads. Considering the above scenario, we first define a vector,  $U = [U_1^1, U_2^1, U_3^1, U_1^2, U_2^2, U_3^2, U_1^3, U_2^3, U_3^3]$ . We further define

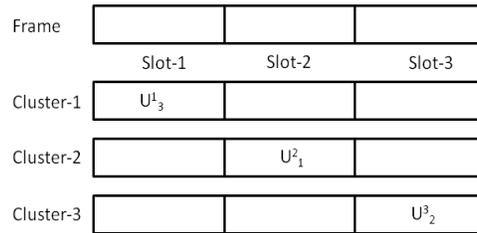


Figure 4.1: Frame Example

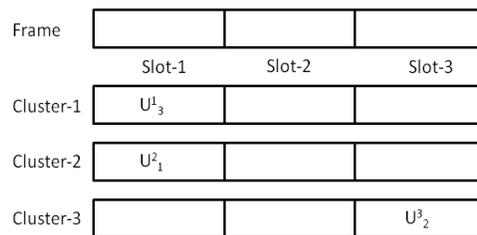


Figure 4.2: Collision in Slots

a matrix  $I$  whose columns are the slot numbers and rows are the replicas of  $U$ , i.e.,

$$\mathbf{I} = \begin{bmatrix} U_1^1 & U_1^1 & U_1^1 \\ U_2^1 & U_2^1 & U_2^1 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ U_3^3 & U_3^3 & U_3^3 \end{bmatrix} \quad (4.7)$$

The entry of this matrix is 1 if that device is assigned in a particular slot and 0 otherwise, i.e.,

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ 0 & 1 & 0 \end{bmatrix} \quad (4.8)$$

shows that device 1 from cluster one is assigned to slot 1, device 3 from cluster three is assigned in slot 2 and device 1 from cluster two is assigned slot 3. Keeping in view the Figures 4.1, 4.2 and Equations (4.7), (4.8), the problem of assigning devices in each slot while avoiding collisions at the same time can be formulated as finding a matrix  $I$  according to the following constraint:

$$|I^1|, |I^2|, |I^3| = 1 \quad (4.9)$$

$$|I_1|, |I_2|, |I_3| \leq 1 \quad (4.10)$$

$$|U_1^1, U_2^1, U_3^1|, |U_1^2, U_2^2, U_3^2|, |U_1^3, U_2^3, U_3^3| = 1 \quad (4.11)$$

where  $I^i$  is the  $i^{th}$  column and  $I_j$  is the  $j^{th}$  row of  $I$  and  $||$  represents the number of non-zero elements. The first constraint ensures that each slot is occupied, the second constraint ensures that each cluster assigns at least one device to each slot, while the third constraint ensures that each cluster does not take more than one slot.

In a general case where there are  $T$  slots,  $K$  clusters and  $u$  devices, the problem can be defined as finding a matrix  $I$  of size  $(u \times K) \times T$  as:

$$|I^i| = 1, \forall i = 1, 2 \dots T \quad (4.12)$$

$$|I_j| \leq 1, \forall j = 1 \dots u \times K \quad (4.13)$$

$$\sum_{k=1}^u U_k^l = 1, \forall l = 1, \dots K \quad (4.14)$$

The solution to this matrix is found by using the Q-learning algorithm, i.e., unique slots are determined by controllers with repeated experience. Data transmission is ensured after the favourable slot selection avoiding collision and resulting congestion.

### 4.3.2 Multiple Frequency Case

Now we consider the case in which multiple frequencies are used in each cluster. Let  $w = (1, 2 \dots W)$  be the number of frequencies in a cluster and MTCD  $i$  in  $j^{th}$  cluster using  $w^{th}$  frequency channel be denoted as  $U_i^{j,w}$ . Consider the same scenario as for single frequency case, and vector  $U$ , matrix  $I$  are modified for multiple frequencies as follows.

We first define a vector,  $U_w = [U_1^{1,w}, U_2^{1,w}, U_3^{1,w}, U_1^{2,w}, U_2^{2,w}, U_3^{2,w}, U_1^{3,w}, U_2^{3,w}, U_3^{3,w}]$  that shows the  $i^{th}$  device in  $j^{th}$  cluster using frequency  $w$ . We further define a matrix  $I$ , which is a concatenation of matrices for each frequency. Each matrix represents a single frequency and has columns as the slot numbers and rows as the replicas of  $U$  at a single frequency  $w$ , i.e.,

$$I = [I_{w=1}, I_{w=2}, I_{w=3}] \quad (4.15)$$

$$I_w = \begin{bmatrix} U_1^{1,w} & U_1^{1,w} & U_1^{1,w} \\ U_2^{1,w} & U_2^{1,w} & U_2^{1,w} \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ U_3^{3,w} & U_3^{3,w} & U_3^{3,w} \end{bmatrix} \quad (4.16)$$

for each frequency  $w$ . The entry of this matrix is 1 if that device is assigned to a particular slot at the frequency  $w$  and 0 otherwise, i.e.,

$$\mathbf{I}_w = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ 0 & 1 & 0 \end{bmatrix} \quad (4.17)$$

In this case, our problem can be defined as finding a matrix  $I$  of size  $(u \times K) \times (T \times u)$  by observing the following constraints at each frequency  $w$ :

$$|I^{i,w}| = 1, \forall i = 1, 2 \dots T \quad (4.18)$$

$$|I_j^w| \leq 1, \forall j = 1 \dots u \times K \quad (4.19)$$

$$\sum_{k=1}^u U_l^{k,w} = 1, \forall l = 1, \dots K \quad (4.20)$$

In this section, we showed that the slot allocation problem can be seen as finding a matrix  $I$  under certain constraints that enforce unique slot allocation. In the next section, we shall present a solution for this problem, which is based on the Q-learning algorithm.

## 4.4 Proposed Solution

In this section, we first describe traditional Aloha and channel based allocation techniques. Then we define a Q-learning algorithm for M2M communication followed by a slot selection procedure.

### 4.4.1 Classic Techniques

ALOHA is the most simple and traditional Random Access (RA) technique, in which any user can send any time. We use ALOHA for comparison with our proposed technique. As in our scenario, controllers will transmit data in any slot any time and it is similar to

ALOHA. It may happen that two controllers happen to transmit data in same slot and may result in collision. Channel Based Allocation (CBA) is also performed for comparison, where controllers are allowed to pickup the wireless channel with best channel gain and transmit data in each slot. This criterion accounts for the improved average SNR, but does not prove beneficial for successful channel access. As we are addressing the random access problem, allocation of a collision free channel is the success criterion and above two methods are not much of the benefit in our case. In our future work, we will consider the SNR value for the slot selection, for which CBA seems to be attractive choice. However, it may not prove to be best choice for distributive interference minimization requirement among controllers sharing the same slot.

#### 4.4.2 Reinforcement learning

The basic idea behind reinforcement learning is as follows: an agent learns by trial and error. The agent learns that its actions and decisions have consequences upon the surrounding environment. The agent acts as a controller of this decision process, as the actions that it takes will lead to transitions between states. Upon the selection of an action, the agent should analyze the new state that it has transitioned to. A reward function is defined for each action to check its correctness [137]. A higher reward illustrates that the action had beneficial consequences while a lower one illustrates that a different action needs to be tried out. Let  $R$  be the reward function associated with any action  $A$  and  $Q$  value is updated as:

$$Q_{t+1} = Q_t + \alpha(R - Q_t) \quad (4.21)$$

where  $\alpha$  is the learning rate and  $t$  is the time at which action is taken.

#### 4.4.3 Q-Learning Algorithm for M2M Communication

Q-learning is one of the reinforcement learning algorithms. We are using a Q-learning algorithm for scheduling of MTCs in a clustered network such that there is no collision of

packet transmission. Each cluster head transmits data to its associated devices on a specific frequency and time slot in downlink. Each cluster head tries to randomly transmit to devices, which may result in collision if some other neighbouring controller is also utilizing the same frequency and time slot for transmission simultaneously. Every controller learns with previous experience and schedules its MTCDs in such a way that no two MTCDs of neighbouring clusters are utilizing same slots on the same frequency. Stateless Q-learning [144] is used in this scheme to obtain the learning experience. Each controller has individual  $Q$  values for every slot, and they represent the preference of the slot selections.  $Q$  values are denoted by  $Q(j, T, w)$  and they represent that controller  $j$  takes an action on slot  $T$  for obtaining the frequency channel  $w$ . The previous  $Q$  values and the current reward all contribute to the  $Q$  value update. The  $Q$  value is updated as below after the reward is returned:

$$Q_{t+1}(j, T, w) = Q_t(j, T, w) + \alpha(R - Q_t(j, T, w)) \quad (4.22)$$

- Agent: MTCD controller  $j, \forall 1 \leq j \leq K$ . Controllers (or cluster heads) are the agents running the Q-learning algorithm for the best slot selection.
- Action:  $A(t) = a^{j,T}(t), T \in [T_1, \dots, T_k]$ .  $a^{j,T}(t)$  is defined as the action of  $j^{th}$  controller at time  $t$  and is to choose time slot  $T_j$  out of  $T$ , where  $T_k$  is the maximum number of time slots in a frame. Controllers will transmit data on the specific slots according to some information obtained in the previous slots.
- Reward:  $R$  is defined as:

$$R = \begin{cases} 0 & T_j = \phi, \\ +1 & T_j \text{ is unique,} \\ -1 & T_j = T_{\hat{j}}. \end{cases} \quad (4.23)$$

where  $T_j$  is the slot chosen by the  $j^{th}$  cluster head and  $T_{\hat{j}}$  is the same slot chosen by any other cluster head.

Each controller is using the Q-learning algorithm to transmit data randomly in time slots. Also, the ACK/NACK protocol is being used by these controllers to get notified for the success/failure of the transmission in a given slot. If no two cluster heads select the same slot, the transmission is successful and ACK is received. Thus a controller will assign a positive reward value to that slot. If the transmission fails, i.e., two (or more) controllers send data in the same time slot resulting into collision and NACK is received by both (or more). And a negative reward is assigned to that slot. In this way, controllers will learn by experience which slot is good for data transmission by comparing their  $Q$  values. Slots with higher  $Q$  values will always be preferred, i.e.,

$$I^{j,w} = \max_{T,j,w} Q_t(j, T, w) \quad (4.24)$$

If multiple slots have the same  $Q$  value, controllers will randomly select one of them. All the  $Q$  values are initialised to 0 at the very beginning. Figure 4.3 shows the flow of operation of controllers finding the best slot for data transmission with the Q-learning algorithm.

In a multiple frequency case,  $w$  frequencies are available in a cluster for use by MTCDs. For simplicity of analysis, we keep  $u = w$  in a cluster such that each MTCD will get a unique frequency, thus avoiding intra-cluster interference. We assign the first frequency to MTCD 1 of each cluster, second to MTCD 2 and so on. Another method of channel assignment can be based on the best channel for each MTCD and cluster controller. The proposed Q-learning algorithm will run for each frequency in a cluster controller to find the appropriate slot allocation.

In this section, we showed that a solution to the problem of slot allocation can be seen as maximization of Q-values among different slots. As the algorithm is dependant on  $\alpha$  and  $R$ , we shall present a thorough analysis in the next section that shows how the performance is affected by these parameters.

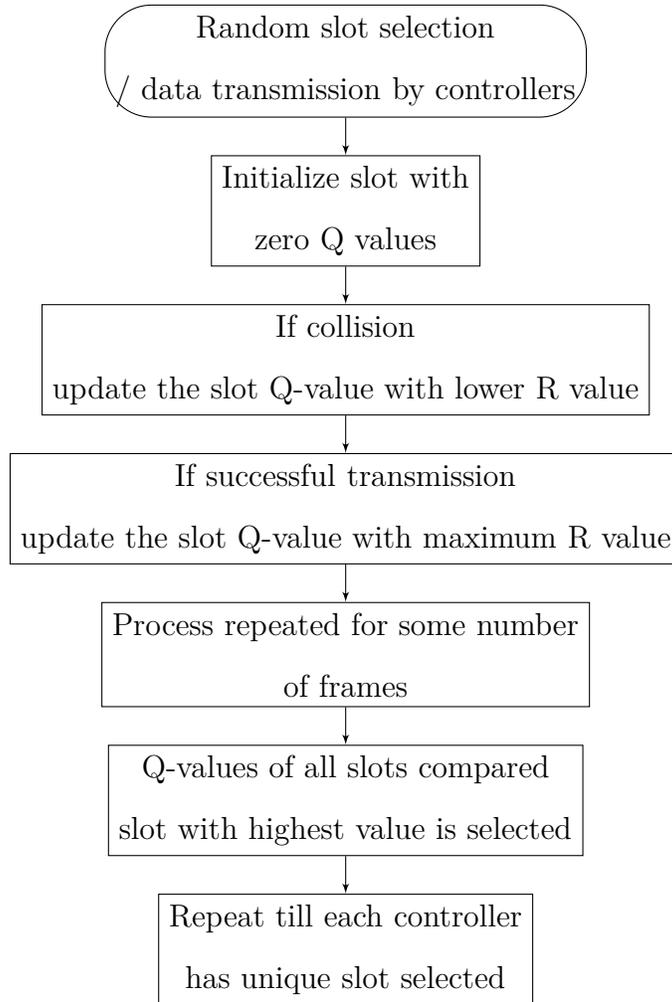


Figure 4.3: Slot Search Flow Chart

## 4.5 Results and Discussions

In this section, we present simulation results showing the performance of the proposed algorithm. We compare the convergence time with respect to varying learning rate ( $\alpha$ ) and reward ( $R$ ). We further compare the cumulative success probability with respect to channel based selection (scheduled) and ALOHA (totally random) in channel assignment.

The simulation setup is as follows: We consider four clusters ( $K = 4$ ), having three devices ( $u = 3$ ) in each, as shown in Figure 4.4. Each cluster has three frequencies ( $w = 3$ ) controlled by the cluster head. A frame also has four time slots ( $T = 4$ ), one slot is for each

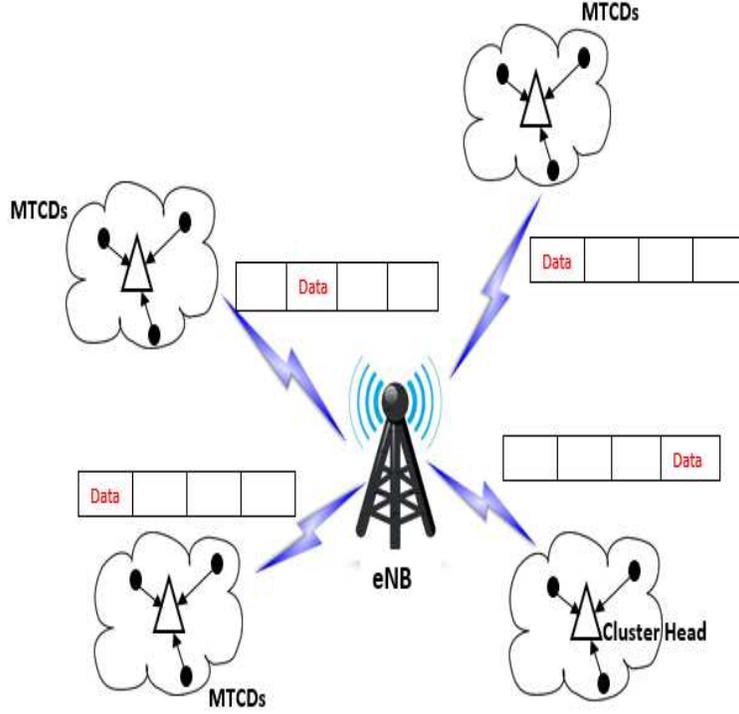


Figure 4.4: Clustered M2M Network

controller. Each controller will transmit data coming from eNB to its associated devices in one of the selected time slots using the proposed Q-learning algorithm. For ALOHA, controllers will randomly select a slot and transmit data to its associated devices. It will result into collision if two or more controllers choose to send in the the same slot. After collision, controllers will retransmit again by randomly picking the slot. For the channel based allocation, controllers are allowed to pickup the channel with best channel gain, without considering the selection made by neighbouring controllers. It may result in collision if the same slot is selected by others for the same channel. This transmission is considered unsuccessful and retransmission takes place following the same procedure.

#### 4.5.1 Effect of Learning Rate $\alpha$

We assign different values of  $\alpha$  and variable positive and negative reward values for  $R$ . In Figure 4.5, convergence time is analyzed with respect to learning rate. We define it as the

time taken by the controllers to learn with repeated data transmission and resulting  $Q$  values, for the selection of a unique time slot for their data transmission. Convergence is declared when there is no change in slot assignment. We calculate convergence time obtained over 1000 iterations and average the results. Different values of  $\alpha$  varying from 0.1 to 0.9 in a step-size of 0.1 are used to study the rise and fall of the convergence rate. The convergence rate varies from 4 to 13. It is noted that as  $\alpha$  is increasing, the convergence rate increases and it takes less time to converge. This is an expected behavior as with an increase in  $\alpha$ , more weight is given to the reward than to the current  $Q$  value. This can be seen by rewriting (4.22) as follows:

$$Q_{t+1}(j, T, w) = Q_t(j, T, w)(1 - \alpha) + \alpha R \quad (4.25)$$

We can see that by increasing  $\alpha$ , the ratio of weight assigned to  $R$  compared to  $Q_t(j, T, w)$  is given as  $\frac{\alpha}{1-\alpha}$ . Increasing value of  $\alpha$  gives more weight to  $R$ . As a result, quicker convergence is achieved due to more emphasis given to current values and actions. For  $\alpha = 0.1$  to 0.9, the ratios are given as:  $\frac{1}{9}, \frac{1}{4}, \frac{3}{7}, \frac{2}{3}, 1, \frac{3}{2}, \frac{7}{3}, 4, 9$ , respectively. Hence, as the ratio increases, the convergence time decreases.

Figure 4.6 shows convergence probability versus  $\alpha$ . Convergence probability is defined as ratio of number of times simulation converge and the total number of simulations, i.e. 1000.  $\alpha$  is varied from 0.1 to 0.9 in a step size of 0.1. Except for a small increase in convergence probability between 0.1 and 0.2, a decreased trend can be observed. From Figures. 4.5 and 4.6, we can observe that a higher  $\alpha$  decreases both convergence time and probability. Thus a lower  $\alpha$  is preferable for increased convergence probability at the expense of increased convergence time.

### 4.5.2 Effect of Reward $R$

We now show the effects of varying both positive and negative rewards on the the average convergence time. In Figure 4.7, negative and positive rewards are kept the same at one instant. Analysis is done for different values of reward function i.e, at  $\pm 0.1, \pm 0.2, \pm 0.3$  and

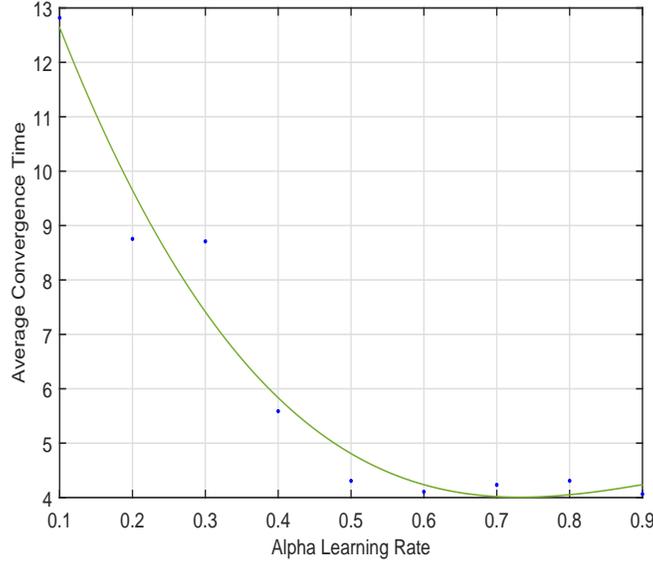


Figure 4.5: Variable Learning Rate; dots show the actual data points and line shows the curve fitted data

$\pm 0.4$ ,  $\pm 1$ ,  $\pm 2$ ,  $\pm 3$  and  $\pm 4$ . The value of  $\alpha$  for the plots is fixed as 0.1 and 0.9. Convergence time for different reward values is shown. It can be seen that for  $\alpha = 0.1$ , the convergence time decreases when the reward is changed from  $\pm 1$  to  $\pm 2$ , and stays almost the same when the reward is  $\pm 4$ . For  $\alpha = 0.9$ , the convergence time is constant with respect to changing reward. The reason for this behavior can be described as follows.

By increasing the reward, more emphasis is placed on the current value of reward  $R$ . This fact can be seen by considering that if the reward is increased by a factor  $f$ , (4.22) can be rewritten as:

$$Q_{t+1}(j, T, w) = Q_t(j, T, w) + \alpha(f \times R - Q_t(j, T, w)) \quad (4.26)$$

then can be further written as:

$$Q_{t+1}(j, T, w) = Q_t(j, T, w)(1 - \alpha) + f \times \alpha R \quad (4.27)$$

In the above equation, the ratio of weight assigned to  $R$  compared to  $Q_t(j, T, w)$  is given as  $\frac{f\alpha}{1-\alpha}$ . As the value of  $f$  is increased, the ratio increases meaning that more weight is given

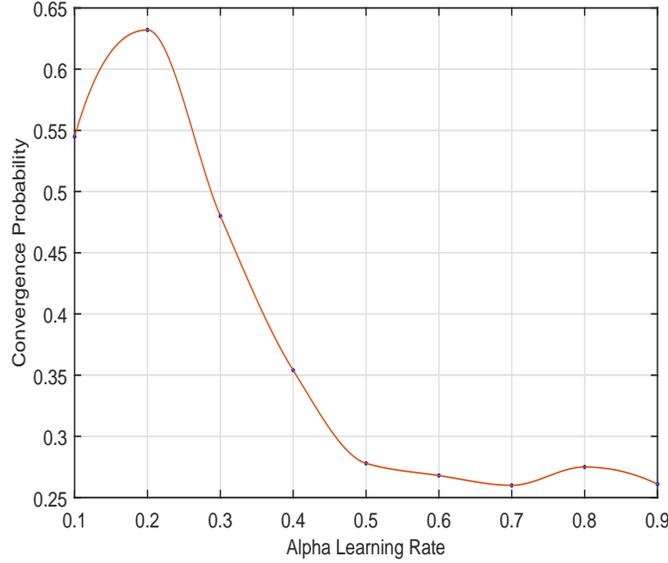


Figure 4.6: Convergence Probability

to  $R$ .

Let us consider  $\alpha = 0.1$  and  $0.9$ . In this case, the ratio of weights assigned to  $R$  compared to  $Q_t(j, T, w)$  is given as  $\frac{f \times 0.1}{0.9}$  and  $\frac{f \times 0.9}{0.1}$ , respectively. Now the reward is changed from  $f = 1$  to  $f = 4$ , this ratio is given as  $0.11, 0.22, 0.33$  and  $0.44$ , respectively. For  $\alpha = 0.9$ , and this ratio is given as  $9, 18, 27, 36$ , respectively. When the reward is changed from  $f = 0.1$  to  $f = 0.4$ , this ratio is  $0.011, 0.022, 0.033$  and  $0.044$ , respectively, for  $\alpha = 0.1$ , and  $0.9, 1.8, 2.7$  and  $3.6$ , respectively, for  $\alpha = 0.9$ .

Hence, as the ratio is increasing, the convergence time decreases for  $\alpha = 0 - 1$ . For  $\alpha = 0.9$ , the ratio is already so high that changing it does not have much effect on the convergence time, hence a nearly constant trend is seen. When the reward  $> \pm 2$ , trend almost becomes constant due to higher values of weight, as can also be seen in Figure 4.5.

In Figure 4.8, negative reward is fixed at  $-1$  and the positive reward is varied, whereas in Figure 4.9, positive reward is fixed at  $+1$  and the negative reward is varied. It can be seen that as the positive reward is increased, the convergence time decreases for  $\alpha = 0.1$  and stays the same for  $\alpha = 0.9$ . In Figure 4.9, as the magnitude of negative reward is decreased,

the convergence time decreases for  $\alpha = 0.1$  and stays almost constant for  $\alpha = 0.9$ , which shows that that magnitude of negative reward should be less, as then, collision will have less effect.

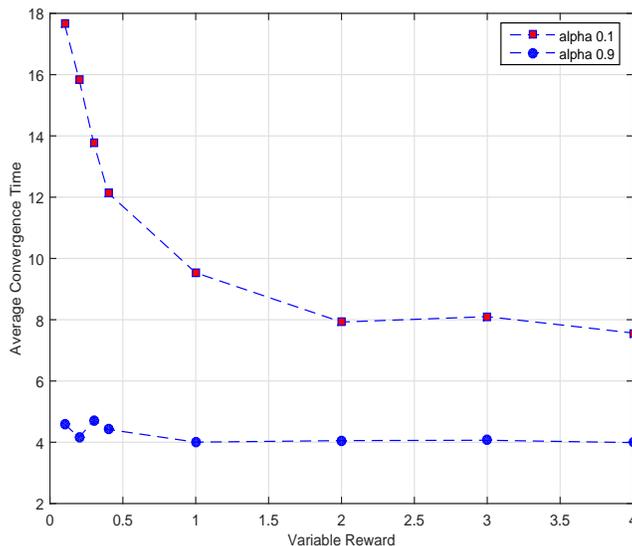


Figure 4.7: Variation in Both Negative and Positive Rewards

### 4.5.3 Effect of Both $\alpha$ and $R$

Relationship between alpha and positive/negative reward is further investigated through Figures. 4.10 and 4.11. In Figure 4.10, relationship between one positive and four negative reward values at one instance is investigated. Four curves in this figure shows four different positive rewards, and their relationship with all four negative reward values at  $\alpha = 0.1$ . Same plots are obtained for  $\alpha = 0.9$ .

In Figure 4.11, relationship between one negative and four positive reward values at one instance is investigated. Four curves on this figure show four different negative rewards and their relationship with all four positive reward values for  $\alpha = 0.1$ . Same plots are obtained for  $\alpha = 0.9$ . A trend similar to Figures 4.7, 4.8 and 4.9 can be seen. For one value of positive reward, the convergence time decreases when the negative reward is decreased, or

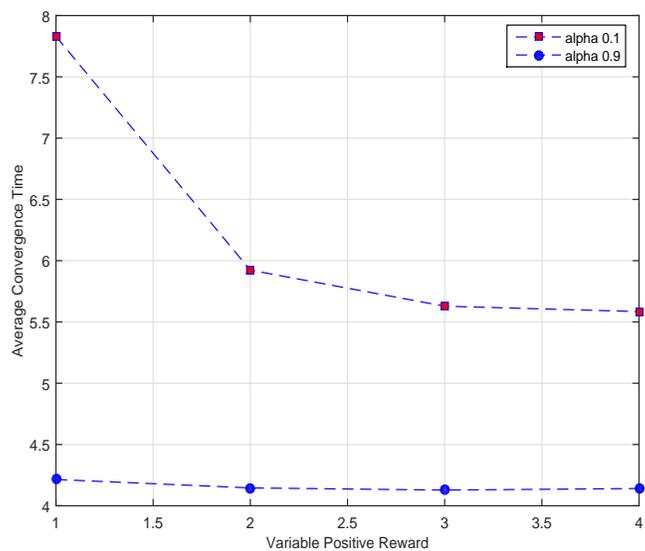


Figure 4.8: Variable Positive Reward

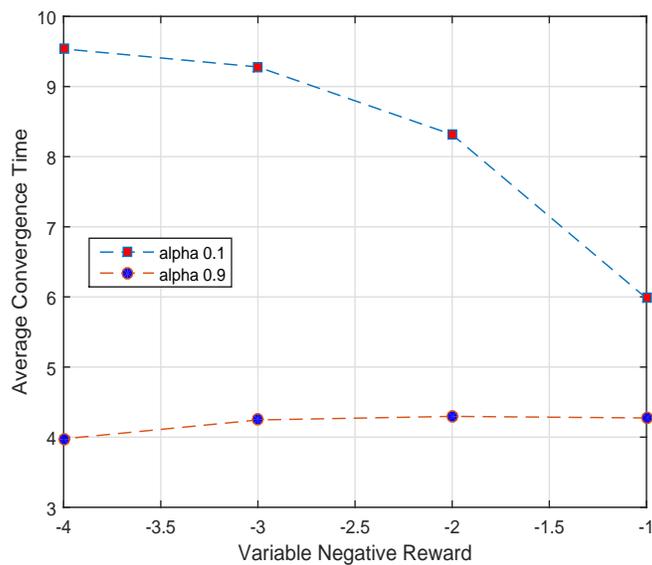


Figure 4.9: Variable Negative Reward

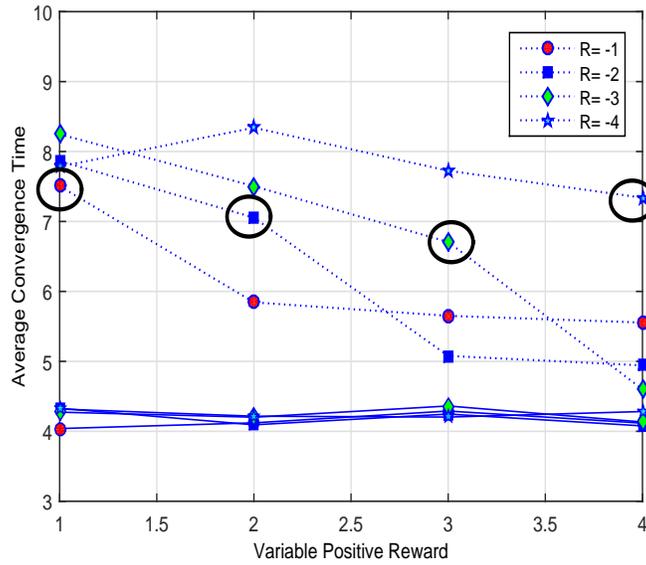


Figure 4.10: Variation in Positive Reward with Different Negative Rewards at  $\alpha = 0.1, 0.9$ .  
dotted line:  $\alpha = 0.1$ , solid line:  $\alpha = 0.9$

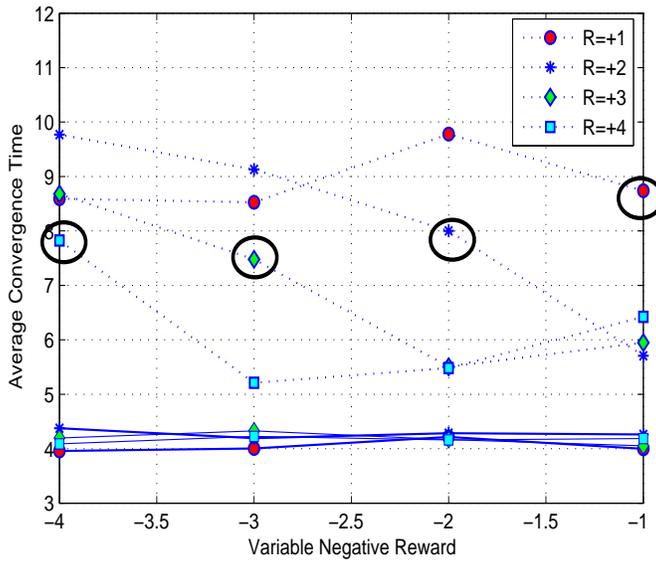


Figure 4.11: Variation in Negative Reward with Different Positive Rewards at  $\alpha = 0.1, 0.9$ .  
dotted line:  $\alpha = 0.1$ , solid line:  $\alpha = 0.9$

for one value of negative reward, the convergence time decreases when the positive reward is increased. Some variations from this trend may be noted; however, it can be observed from the encircled points that the convergence time shows little variation when both rewards are the same in the range of one to four. This can also be observed from Figure 4.7, which further confirms our analysis.

#### 4.5.4 Q-Learning versus RA/CBA

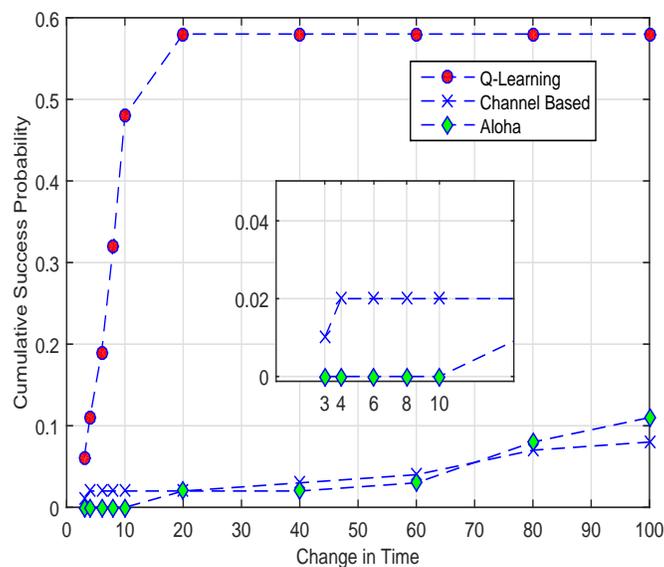


Figure 4.12: Comparison between Q-Learning and Channel Based Allocation

In Figure 4.12, comparison is made among ALOHA type random access (RA), channel based access (CBA) and the proposed Q-learning algorithm. ALOHA and CBA are two extreme channel access techniques used as a reference to compare our performance. The cumulative success probability for each is plotted with increasing time. It is defined as the probability of observing less than or equal to a given number of success for slot allocation. Slot selection is done with above mentioned methods. It is seen that Q-learning performs better than the rest as expected. Q-learning is able to achieve a cumulative success probability of 0.6, whereas for the other two approaches, it is around 0.1. As with Q-learning, each

cluster head learns with experience which slot is better in terms of data transmission and reduced collisions, resulting in better success rate. CBA shows slightly better performance than ALOHA, but poor performance values compared with Q-learning plot. For Q-learning, the probability increases almost linearly till 20 slots and becomes constant after that. This constant value indicates the convergence point for the Q-learning algorithm. Once converged, i.e., once each controller learns by experience to successfully transmit in specific time slot without any collision, it will continue to transmit into the same slot.

In this section, we showed the dependency of our solution on  $\alpha$  and  $R$  values and showed that our algorithm has higher success probability. Based on simulation results and analysis, we can make the following conclusions:

- A lower  $\alpha$  value gives higher convergence time and higher convergence probability.
- Increasing both positive and negative rewards can be seen equivalent to increasing the learning rate.
- A higher positive reward and lower magnitude of negative reward gives lower convergence time.
- Equal positive and negative rewards give low variation in the convergence time for a reward greater than or equal to one.
- For lower values of  $\alpha$ , changing the reward significantly affects the convergence time.

## 4.6 Chapter Summary

In this chapter, we presented a solution to the slot assignment problem for machine type communication devices (MTCDs) in M2M communication. We first used a clustering algorithm to overcome the congestion problem. Then we showed that the slot allocation problem can be seen as a constrained optimization problem. We used a Q-learning algorithm to solve this problem. The dependency of the solution on learning rate and rewards is subsequently

analyzed. We showed that when the learning rate increased, the convergence time decreased and the convergence rate increased. We demonstrated by means of analysis that the effect of increasing reward values is the same as increasing the learning rate. Simulation examples demonstrated that the convergence time decreased with the increase of positive reward values and with a negative reward with lower magnitude. We further showed that the negative and positive rewards should be similar in order to have a consistent behavior. We compared our technique with ALOHA and channel based assignment and showed that our method had better performance. The cumulative probability of our solution successfully assigning slots is more than five times higher compared to these methods. For the future work, we aim to use Q-learning within a distributive shared slot TDMA environment in the next chapter.

In the next chapter, we present a block based Q-learning algorithm for the scheduling of MTCDs in distributive manner in clustered M2M communication networks. At first centralized slot assignment is done and an algorithm is proposed for minimizing the inter-cluster interference. Then we use Q-learning algorithm to assign slots in a distributed manner and comparison is made between the two schemes.

# Chapter 5

## Slot Allocation with Interference

Earlier slot allocation for MTCs is considered with no interference amongst transmissions. However, this is inefficient as frequency/time resources are not reused spatially. With some tolerable interference (subject to SIR threshold level) the capacity of M2M system can be improved, which is the focus of the work in this chapter.

In this chapter, we present a block based Q-learning algorithm for the scheduling of machine type communication devices in clustered M2M communication networks with inter-cluster interference. We first form clusters based on the spatial distribution of MTCs. At first centralized slot assignment is done and an algorithm is proposed for minimizing the inter-cluster interference. Then we use Q-learning algorithm to assign slots in a distributed manner and comparison is made between the two schemes. Afterwards, we show the effects of distributed slot-assignment with respect to varying signal-to-interference ratio on convergence rate and convergence probability. Cumulative distribution function is used to study the effect of various SIR threshold levels on the convergence probability. With the increase in SIR threshold levels, increase in convergence time and decrease in convergence probability are observed, as less block configuration fulfills the required threshold in the M2M network.

## 5.1 Introduction

In a typical machine to machine (M2M) network, large number of MTCs exist and they mostly transmit a small amount of data. This results in simultaneous random access of channel resources, which causes overload and access problem. Clustering is a technique that can reduce network congestion and increase the energy efficiency [125,126]. In [33], managing radio resources over clusters instead of individual MTCs and the corresponding benefits are presented. In our work also, we implement the clustering of MTCs with cluster heads (or controllers) to facilitate the hierarchal control allocation of resources in massive M2M networks.

We first consider the centralized slot allocation for MTCs in our clustered network followed by the distributed assignment. We perform the comparison of the two and conclude the superiority of the latter in terms of MTC admission rate. Also, distributed allocation performs well in terms of computational complexity and time, reduced network traffic and no involvement of evolved node base station (eNB).

In [145], resource allocation problem is analyzed through both non-cooperative and cooperative game to maximize their data rate and minimize utilization of power. The performance of coordinated and uncoordinated transmission strategies for multiple access is analyzed in [133,134]. Authors in [146] proposed a predictive resource allocation scheme employed at the eNB based on propagation characteristics for M2M applications. Similarly resource allocation and scheduling is discussed in [52,64]. In [147], classification of M2M scheduling techniques from the perspective of versatile traffic requirements is presented.

Q-learning algorithm is a well known reinforcement learning (RL) technique, which enables a MTC in a M2M network to learn by interacting with its environment [148]. Afterwards, it adapts to the best action by gaining experience with number of trials [11,149]. Authors in [150] propose an online RL algorithm that continuously adapts to the changing network traffic in deciding which action to take to maximize energy saving at eNB. A decentralized Q-learning technique is proposed in [151] to manage the interference generated by

multiple wireless area networks. Q-learning is employed in [152] to choose different transmission parameters and to make an efficient assignment of spectrum and transmit powers in cognitive radios. In [153], the RL approach is used for interference mitigation of a macrocell network underlaid with self-organized femtocells, in which each femtocell adapts its strategy and gradually learns by interacting with its environment. In [154], a distributed Q-learning algorithm for sharing spectrum among femtocells and macrocells in a decentralized manner is presented. In [155], online reinforcement learning framework is presented for the traffic offloading in a stochastic heterogeneous cellular network.

Fast adaptive slotted Aloha (FASA) is proposed in [139] taking into account the knowledge of the idle/successful/collided state of the previous slots. This information is exploited to improve the performance of the access control protocol. Slotted ALOHA exploiting successive interference cancellation (SIC), called Frameless ALOHA, is presented in [140]. These techniques can ideally guarantee high performance in a M2M scenario in terms of throughput. Nonetheless, energy efficiency and complexity aspects are not considered. In particular, the SIC mechanism sets quite high requirements to the eNB in terms of storage and processing capabilities. In addition, the energy consumption of MTCs' is a major concern here as for each frame, the devices must transmit a large set of replicas of the same packet to the eNB.

In [54], RL-based base station selection algorithm is proposed that allows the MTCs to choose the base station in a self-organizing fashion. While [137] used QoS performance measure to switch from one base station to another and it is the ratio between the device throughput and its delay. In [138], a distributed algorithm in which MTCs share resources with a particular cellular user in a TDMA manner was proposed.

Q-learning RACH access scheme (QL-RACH) is proposed in [141, 142] to control M2M traffic in order to reduce its impact on a cellular network. It uses ALOHA and an intelligent slot assignment strategy to avoid collisions amongst the MTCs. Q-learning based MAC with informed receiving (ALOHA-QIR) for wireless sensor networks (WSNs) is presented in [156], where frame based slotted ALOHA and Q-learning are used such that nodes have

certain intelligence to access slots with a lower probability of collision. However, the performance of these scheme is dominated by the uncontrolled of H2H traffic especially at the upper load limit. Also, there is a restriction of using a virtual M2M frame having length (in time slots) equal to the number of MTCs in the network. This imposes the upper bound on the number of MTCs being served simultaneously. In addition to this, every node keeps a Q-value for each slot in the M2M frame to record the transmission history on that slot in consecutive frames, which is not an energy efficient mechanism for the energy constrained, battery limited nodes.

In this chapter, we try to avoid overload and congestion by forming clusters on the basis of MTCs spatial distribution. Afterwards, inter-cluster interference is minimized for meeting the desired signal to interference ratio (SIR) by the MTCs in each cluster. We deal with the problem of slot assignment in random access network (RAN) of MTCs in a clustered network and present a strategy for selection of time slots in a frame by controllers in TDMA-based M2M networks. As same time slots are used by all the controllers, there will be inter-cluster interference. Our purpose is to assign the MTCs in each cluster distributively in such a way that inter-cluster interference is minimized. In addition to this, the concept of Q-learning is applied on a block of slots instead of individual slots.

In our previous work [18], distributed channel assignment in clusters was done avoiding inter-cluster interference. However, it is inefficient because one slot was being utilized by only one cluster and no spatial reuse of time slots was realized. In this work, slots can be used simultaneously by more than one cluster however giving rise to interference. But this interference is minimized by the use of block-based Q-learning algorithm in this paper. Our work in comparison with [141, 142, 156] is novel in a sense that we reduce the computational complexity and time by introducing the concept of block-based or clustered slot allocation using Q-learning. We improve the efficiency by performing distributed slot assignment, by not only minimizing inter-cluster interference, but also benefitting from frequency reuse among all the clusters. We do not study the channel allocation strategies in this work. It is expected that frequency diversity will increase the SIR performance in a multi-channel

system.

The contribution of our work is the performance enhancement of MTCs through selecting the minimum conflict time slots in congested scenarios using Q-learning.

- MTCs are arranged into clusters to overcome the congestion and overload problem in M2M network. Each MTC is associated with the nearest controller with the help of K-means algorithm. As a result, each MTC has to use less energy to send its data to respective controller resulting in an energy efficient system.
- A centralized algorithm is proposed for slot allocation for MTCs such that minimum inter-cluster interference is experienced by all the devices. This is based on graph colouring mechanism.
- Slot allocation based on the Q-learning algorithm is proposed. Block based or clustered slot allocation for M2M communication in a distributed manner is presented.
- Comparison between the centralized and distributed slot allocation is made in terms of average SIR and admittance rate is made, for the clustered M2M network.
- Computational complexity and time is improved by introducing block-based allocation using Q-learning.
- We further analyze the convergence capabilities ( convergence time and convergence probability) of the proposed approach with respect to different parameters.
- We propose various reward schemes for the Q-learning algorithm and perform the comparison.

This chapter is organised as follows: Section II describes the system model and section III presents centralized and distributed methods for resource allocation. It introduces an idea of block Q-learning and presents it as a preferred solution for resource allocation. Simulation results are presented in section IV and section V concludes this chapter.

## 5.2 System Model

A clustered network with downlink communication is considered as shown in Figure 5.1. We assume that number of MTCs per unit area is high and a single eNB is not able to serve all the devices in an efficient manner. A single eNB is present that is in communication with the cluster heads/controllers. MTCs controllers are responsible for data aggregation and transmission to their respective MTCs as shown in the figure. When more than one controller transmits in the same time slot, interference is experienced as the transmission is not coordinated.

We consider a TDMA-based network with a frame having  $T$  slots and a controller for each cluster will send data to its associated devices in these slots. Let  $K$  be the total number of clusters and  $D$  be the maximum number of devices in each cluster.

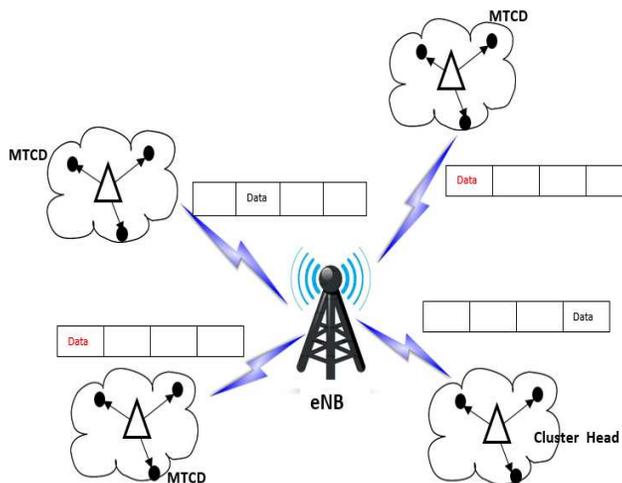


Figure 5.1: Clustered M2M Network

Each slot in each cluster will have a device assigned to it for transmission. Let device  $i$  in  $j^{th}$  cluster be denoted as  $U_i^j$ , e.g., in the Figure 5.2, slot 1 in cluster 1 has user 1 assigned to it. Slot 1 in cluster 2 has also user 1 assigned to it, whereas slot 1 in cluster 3 has user 3 assigned to it. Let the  $r^{th}$  frame be denoted as  $F_r$ , e.g., frame 1 in cluster 1 is

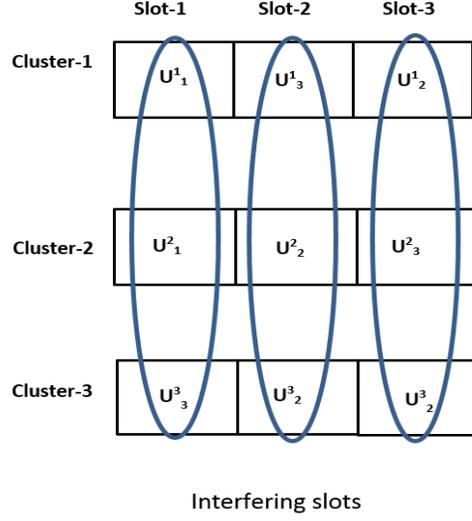


Figure 5.2: Frame Allocation and Interference. ( $T = 3, K = 3, D = 3$ )

$F_1 = [U^1_1, U^1_3, U^1_2]$ . Similarly,  $F_2 = [U^2_1, U^2_2, U^2_3]$  and  $F_3 = [U^3_3, U^3_1, U^3_2]$  are frame 1 of cluster 2 and cluster 3, respectively in the previous example.

There is no intra-cluster interference as TDMA is being used within the cluster and all devices are sharing the same frequency but at different times. However, as each time slot is used by all the controllers (or clusters), there will be inter-cluster interference. Our purpose is to assign slots to the users in each cluster in such a way that inter-cluster interference is minimized with reinforcement learning approach such as Q-learning.

Let the interference between different clusters in slot  $t$  be denoted as  $E_t$  on a particular frequency channel. This interference is the function of different devices assigned to slot  $t$  in each cluster. In the case of Figure 5.2, devices 1, 1, 3 are assigned to slot  $t = 1$  in cluster 1, 2, 3, respectively. Let  $\gamma_t$  be the resulting signal to interference ratio (SIR) of devices in slot  $t$ . The SIR can be written as:  $\gamma_t = f(U^1_1, U^2_1, U^3_1)$ , where  $f = ()$  is the function that calculates the SIR based on MTCs assignment. We calculate the actual power transmitted by each controller to its associated devices based on distance and the same received power, which is used to calculate the interference caused to devices sharing the same time slot.

Afterwards, we calculate the SIR for each device in all the clusters. We want to find the combination of devices assigned to slot  $t$  in such a way that the interference is as low as possible and SIR is maximum. In the above example with three clusters, it can be written as:

$$\begin{aligned} & \max_{U_i^j} \gamma_t, \forall t, i = 1 - 3, j = 1 - 3 \\ & \text{subject to} \\ & |F_r| = T, \forall r \end{aligned} \tag{5.1}$$

where  $F_r$  counts the number of devices assigned in  $r^{th}$  frame. The above constraint means that all devices are assigned slots in all the frames and no frame should be left empty. In a general case where there are  $T$  slots,  $R$  frames,  $K$  clusters and  $D$  MTCDs, the problem can be written as: At each time instant  $t$ ,

$$\max \gamma_t, \forall t = 1, 2, \dots, T \tag{5.2}$$

$$\text{s.t. } |F_r| = T, \forall r = 1, 2, \dots, R \tag{5.3}$$

$$U = [U_1^1 \dots U_D^1, U_1^2 \dots U_D^2, \dots, U_1^K \dots U_D^K] \tag{5.4}$$

where  $U$  is the vector consisting of all the devices in each cluster. Here, each MTCD is numbered from 1 to  $D$ , and we assume equal number of devices in each cluster.

### 5.3 Proposed Solution

There are two ways for slot assignment; centralized and distributed. In the centralized method, eNB is responsible for the formation of conflict graph and it broadcast the conflict free slot assignment to all the controller/heads. We use graph colouring approach to identify slot assignment to MTCDs. Afterwards this slot schedule is advertised by the controller to all the MTCDs in its cluster. While in the distributed slot assignment, each controller is responsible for the formation and advertisement of the slot schedules.

### 5.3.1 Centralized Slot Allocation

Downlink data transmission from controllers to their respective MTCDs is analogous to many independent point-to-point flows in the network. Best scheduling assignment is the conflict free assignment which is some times not possible due to existence of large number of MTCDs per unit area (as a distributed assignment). We try to assign slots in such a manner that all MTCDs are able to maintain the required SIR threshold. The conflicting node transmissions are determined based on an interference graph in the centralized assignment.

The network is represented by a graph  $G = (V, E)$ , where  $V$  is the set of nodes in a graph  $G$ . These nodes correspond to  $D + K = |V|$ , total number of MTCDs and controllers in a network. The  $E$  are the transmission links from controllers to MTCDs to be scheduled).

A node may interfere/overhear another node, so these nodes should not transmit/recieve simultaneously. The interference graph  $C = (V, I)$  is found by the estimation of SIR level among various nodes.  $I \subset V \times V$  is the set of edges such that  $(i, p) \in I$ , and  $i$  and  $p$  belongs to  $n$  and  $m$  clusters respectively, if either  $i$  or  $p$  can hear each other or one of them can interfere with a signal intended for the other (even if they cannot hear each other). Therefore, if  $i$  is receiving,  $p$  should not be scheduled to receive from controller at the same time. Therefore, any conflicting nodes are not colored the same color.

Coloring a graph (nodes) is analogous to assigning a time slot to various MTCDs. Controller corresponding to each cluster should take into account the interferers within their range while generating schedules. After assigning the colors to the nodes, controllers broadcast this information to neighboring controllers. Neighboring controllers assign the time slots to the nodes associated to them, taking into account the already assigned potentially conflicting nodes, which are associated with previous controllers. Table 5.1 states the steps for the slot assignment algorithm. Let us consider  $i$  MTCD served by cluster head  $n$  and  $G_{ni}$  be the channel gain between them.  $P_{ni}$  be the transmission power of cluster head  $n$ .

Table 5.1: Algorithm for SIR Based Slot Assignment

1) At start time slots are assigned to all the MTCs of a biggest cluster.
2) Time slots are assigned to all the interfering MTCs of other clusters one at a time.
3) SINR is calculated and time slot giving the best value of SINR is assigned to respective MTC.
4) After the slot assignment is done to all the MTCs in a second cluster, process is repeated for all the clusters in the network.

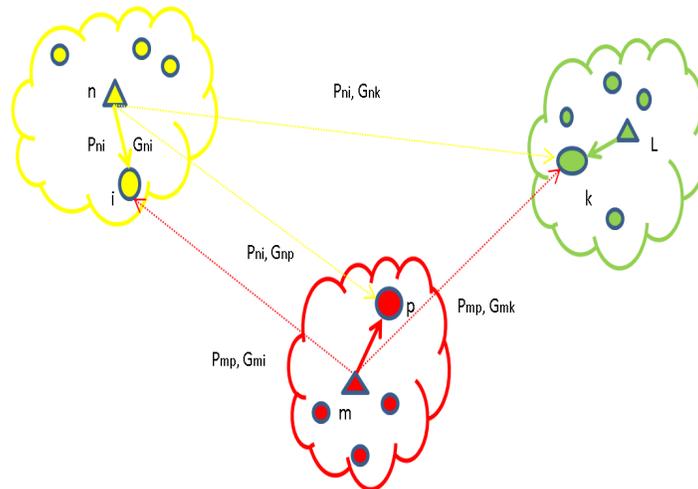


Figure 5.3: Interference Graph

Expression for the SIR at  $i$  MTCD is given by (noise is neglected):

$$\gamma_{t(i \rightarrow n)}^i = \frac{P_{ni}G_{ni}}{\sum_{\substack{j=1 \\ j \neq n \\ O \in I_i}}^K P_{jO}G_{jO}}, \quad (5.5)$$

where  $I_i$  is the set of MTCDs in a cell that can potentially interfere with  $i^{th}$  MTCD. Let  $P_{mp}$  and  $P_{lk}$  be the transmitted power from  $m$  and  $l$  controllers intended for  $p$  and  $k$  MTCDs in their own clusters respectively. This transmission is overheard by the  $i^{th}$  MTCD as three of them are using the same time slot  $t$ . Channel from  $m$  and  $l$  controller to  $i^{th}$  MTCD is given by  $G_{mi}$  and  $G_{li}$  respectively. Therefore (5.5) is modified as:

$$\gamma_{t(i \rightarrow n)}^i = \frac{P_{ni}G_{ni}}{P_{mp}G_{mi} + P_{lk}G_{li}} \quad (5.6)$$

Similarly, SIR of MTCD  $p$  served by cluster head  $m$  is given by:

$$\gamma_{t(p \rightarrow m)}^p = \frac{P_{mp}G_{mp}}{\sum_{\substack{j=1 \\ j \neq m \\ O \in I_i}}^K P_{jO}G_{jO}} \quad (5.7)$$

## Slot Assignment

Using an example we assume that total number of slots are 3 in a frame. Twelve MTCDs are served simultaneously by the controller and thirteen onward will experience denial of service. No two MTCDs will have same slot within the cluster but two MTCDs of different cluster share the same slot.

- We start with the cluster of maximum MTCDs and time slots are sequentially assigned to all the members.
- Distance to all the controllers is calculated and search for the potential interfering MTCDs is started from cluster of closest controller.
- Time slots are assigned to  $i^{th}$  MTCD iteratively to calculate the SIR. Time slot giving the best value of SIR is assigned to respective MTCD. From interference perspective,

both  $i$  and  $p$  can safely receive in the same time slot as overhearing caused by them is not significant to obstruct required/proper SIR demands. If it has not been the situation, then both MTCs are declared as interfering neighbours and are assigned different slots.

- After the slot assignment is done to all the MTCs in a second cluster, controller nearest to it is searched and the same process is repeated. This continues for all the clusters till least interference slot assignment for the entire network is done.

### Select Results

We used the centralized algorithm explained above to obtain the conflict free slot assignment as shown in Figure 5.4. Note that we used 8 slots per frame to obtain this figure to better explain the conflict graph and colour assignment. MTCs using the same colour are using the same time slot and these are assigned in a way that minimum interference is experienced by each of them. This colour assignment is listed in Table 5.2.

Simulation results are provided later for centralized algorithm in section IV.

Table 5.2: Colour Assignment

Slot No.	Colour Assigned	Slot No.	Colour Assigned
1	yellow	5	green
2	blue	6	white
3	cyan	7	black
4	magenta	8	red

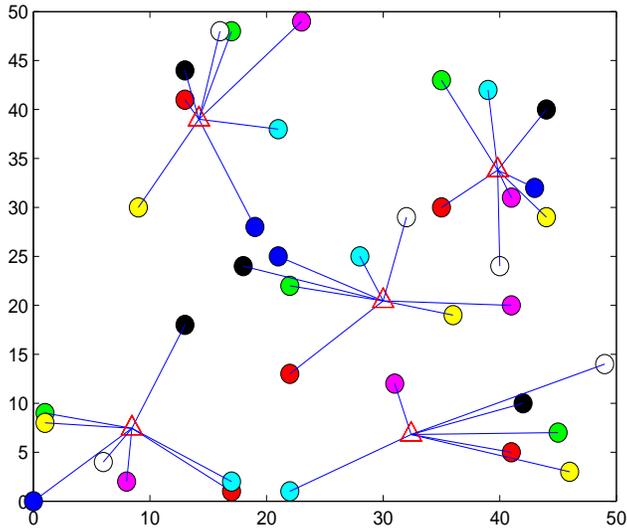


Figure 5.4: Slot Assignment

### 5.3.2 Distributed Slot Allocation

In our problem of slot assignment as a distributed implementation, joint learning and actions are carried out by each controller independent of each other. As controllers are independent of each other and eNB is not involved in any traffic exchange, we call it a distributed approach. Our scenario is the case of multiple agents with multiple actions; therefore, we use Multi-Agent Reinforcement Learning (MARL) algorithm to solve it [157, 158]. There are two distinct ways in which Q-learning could be applied to a multi-agent system. MARL algorithm as an independent learner (IL) algorithm, because the agents perform their actions, obtain rewards and update their Q-values independent of the actions performed by other agents [159]. On the other hand, a joint action learner (JAL) is an agent that learns Q-values by performing joint action and it is also influenced by other agents' actions. In our case, each controller is taking actions independent of each other; therefore, it is MARL with IL. But individual Q-value is affected by other controllers' decisions, therefore, they are also JAL. As there is not any joint Q-value and joint reward update, our problem falls into both

categories. Our problem is identified as distributed, decentralized, MARL with combination of JAL and IL.

### **Multi-Agent Reinforcement Learning (MARL)**

Reinforcement learning (RL) is a mathematical tool for modeling interactions between agents, i.e., machine devices, providing them with the capability of learning, which can enable them to make certain decision [160]. Our case is the multi agent scenario in which each controller is learning from its environment of the suitability of its slot assignment. Each controller transmits data to its associated MTCN on a specific time slot and frequency channel, which are shared by all the controllers. This results in interference caused to one another transmission. Every controller learns with previous experience and schedules its MTCNs in such a way that minimum interference is caused to the neighbouring controller with whom it is sharing the slot and frequency channel.

### **Block Q-Learning for M2M Network**

Block Q-learning or clustered Q-learning is a new concept and introduced for the first time in this paper. As we are considering distributed slot assignment in which controllers are independent of each other decisions. If slot wise learning is performed, complexity increases exponentially with the increase in number of MTCNs, as each device has to learn with respect to every other device in a network. We suggest the block-based assignment in which each controller is assigning slots to all of its associated devices at the same time in contrast to doing it on individual basis. Q-learning is employed to learn with experience which slots are suitable for all the devices in the cluster with a single action for all. This way, not only the computational complexity decreases but also efficiency is improved as each cluster is learning with experience the best suitable slots, influenced by the actions performed by other clusters.

**Block Q-Learning:** It is based on learning Q-experience by controllers on a set or block of slots simultaneously. Reward function is assigned to the block of slots instead of an individual

one, and Q-values are also updated for the block of slots in contrast to the conventional Q-learning usage. Devices are also taken in a group; that is, each controller will assign reward values and gain experience on all the devices at once rather on per device basis. Let  $K(D!)$  be the total number of blocks available to be used by all the controllers, where  $K$  is the number of clusters and  $D$  is the number of devices in each cluster. From this set, each controller will select one block on the basis of previous Q-values and then calculate the reward on the basis of received SIR.

**Complexity:** In conventional Q-learning algorithm, all the learning agents need to maintain Q-functions, one for each slot in the system. These Q-functions are maintained internally by the learning agent, assuming that these values are affected by other agents actions and rewards. In terms of space complexity, it is linear with the frame size. More the number of slots, more is the complexity. Also, as the number of agents increases, computational complexity increases exponentially. We introduced the block-based Q learning in order to reduce the space and computational complexity. space complexity is reduced from  $T^K$  to  $1 \div T_{bl}$  where  $T_{bl}$  is the number of slots in a block. Computational complexity is also reduced from  $K^D$  to  $K(D!)$ , as each controller has to calculate the Q- value and resulting rewards for all the devices simultaneously rather to do it on individual basis.

We define a matrix  $\gamma$ , having rows equal to number of cluster heads and columns as the SIR of each associated device of all controllers. By considering device assignment in Figure 5.2, matrix  $\gamma$  is obtained as under:

$$\gamma = \begin{bmatrix} \gamma_{11} & \gamma_{13} & \gamma_{12} \\ \gamma_{21} & \gamma_{22} & \gamma_{23} \\ \gamma_{33} & \gamma_{31} & \gamma_{32} \end{bmatrix}, \quad (5.8)$$

where  $\gamma_{i,j}$  represents SIR of the  $i^{th}$  device of the  $j^{th}$  cluster. After calculating the SIR of each device, controller will check if it meets the required  $\gamma^{th}$  or not. Using (5.12), '0' and '1' is assigned and matrix  $\hat{\gamma}$  is obtained.

$$\hat{\gamma} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad (5.9)$$

where discrete values for SIRs depend on whether they meet the threshold or not and is assigned in  $\hat{\gamma}$ . For instance, second and third device of cluster 3, and second device of cluster 2 did not meet the required threshold; therefore, 0 is placed. While the rest of the devices in all the clusters met the required threshold, 1 is placed. After obtaining the matrix  $\hat{\gamma}$ , reward  $R^{bl}$  is obtained using (5.16) as explained below.

**Q-Value Update:** Stateless Q-learning is used in this scheme to obtain the learning experience. Each controller has individual Q values for every block, and they represent the preferred block selection such that maximum number of MTCDs is able to meet the required SIR threshold. Q values are denoted by  $Q(j, T_{bl}, w)$  and they represent that controller  $j$  takes an action on block of slots  $T_{bl}$  for obtaining the frequency channel  $w$ . In this paper, we are taking  $w = 1$  and same approach can be used for multiple frequencies. The previous Q values and the current reward all contribute to the Q value update. The Q value is updated as below after the reward is returned:

$$Q_{n+1}(j, T_{bl}, w) = Q_n(j, T_{bl}, w) + \alpha(R_{n+1}^{bl} - Q_n(j, T_{bl}, w)) \quad (5.10)$$

- Agent: MTCD controller  $j, \forall 1 \leq j \leq K$ . Controllers (or cluster heads) are the agents running the Q-learning algorithm for selection of the best block of slots for associated devices.
- Action:  $A(n) = a^{j, \hat{T}_{bl}}(n), T_{bl} \in [T_{bl^1}, \dots, T_{bl^D}]$ .  $a^{j, T}(n)$  is defined as the action of  $j^{th}$  controller at time instance  $n$  and is to choose a block  $\hat{T}_{bl}$  out of  $T_{bl}$ , where  $|T_{bl}| = (D!)$  is the number of blocks available to each controller while  $K(D!)$  is the total number of blocks. For instance, block combinations for controller 1 having three associated

devices are  $3! = 6$ , and they are:

$$T_{bl} = \begin{cases} U_1U_2U_3 \\ U_1U_3U_2 \\ U_2U_1U_3 \\ U_2U_3U_1 \\ U_3U_2U_1 \\ U_3U_1U_2 \end{cases} \quad (5.11)$$

The order of block gives the allocation sequence for each associated device. For example, first block indicates that first device is assigned to first slot, second device to second slot and third device to third slot. These blocks are selected at random in the start followed by the Q-value preference afterwards.

- Reward: is defined as the benefit that controllers will get in terms of SIR, as described below.

**Reward Calculation:** We are assigning reward in two stages:

- First, calculate SIR of individual slots and interpret SIR into discrete values as:

$$R^{sl} = \begin{cases} 1, & \gamma \geq \gamma^{th}, \\ 0, & \gamma < \gamma^{th}, \end{cases} \quad (5.12)$$

where  $\gamma^{th}$  is the required signal to interference ratio by the MTCs for the proper functionality. Using this reward function and (5.9), matrix  $\hat{\gamma}$  is obtained.

- Secondly, actual reward allocation on a block of slots is followed by Q-value update.  $R^{bl}$  is calculated by counting the numbers of 1's in the  $\hat{\gamma}$  as follows for the above example:

$$R^{bl} = \begin{cases} 1 & \text{all } 1's, \\ -1 & \text{one } 0, \\ -2 & \text{two } 0's, \\ -3 & \text{all } 0's. \end{cases} \quad (5.13)$$

**Rewards Schemes:** Three different schemes for reward allocation are suggested.

- **Pessimistic Reward (R1):-** If all the MTCDs of a cluster reach the required SIR threshold, 1 is placed for each slot in a block, and then block reward is assigned as 1. If only 2 devices get the required SIR, block will have one 0 and two 1's, and block reward is assigned as  $-1$ . Hence, for our reward scheme, number of 0's are counted to determine the reward. MTCDs are penalized more for wrong selection of slots and rewarded less for good choice in this scheme. For the given example in matrix  $\hat{\gamma}$ , reward assignment will be as follows:

$$R^{bl} = \begin{cases} 1, \\ -1, \\ -2. \end{cases} \quad (5.14)$$

There are two more reward schemes one can use,

- **Optimistic Reward (R2):-**

$$R^{bl} = \begin{cases} 3 & \text{all } 1's, \\ 2 & \text{two } 1's, \\ 1 & \text{one } 1, \\ -1 & \text{no } 1's. \end{cases} \quad (5.15)$$

- **Balanced Reward (R3):-**

$$R^{bl} = \begin{cases} 1 & \text{all 1's,} \\ 0.5 & \text{one 0,} \\ -0.5 & \text{two 0's,} \\ -1 & \text{no 1's.} \end{cases} \quad (5.16)$$

After obtaining  $R^{bl}$ ,  $Q$  values are updated using (5.10). Controllers will learn by experience which block is good for data transmission by comparing their  $Q$  values. Block with higher  $Q$  values will always be preferred by controller  $j$  for frequency  $w$ , i.e.,

$$I^{j,w} = \max_{T_{bl,j,w}} Q_t(j, T_{bl}, w) \quad (5.17)$$

All the  $Q$  values are initialised to 0 at the very beginning. If multiple blocks have the same  $Q$  value, controllers will randomly select one of them. Figure 5.5 shows the flow chart for block-based Q-learning algorithm used.

All three schemes can be better explained and compared with the example in Table 5.3. We have three clusters and each having three users. Three slots in a frame have to be assigned by controllers to their MTCs in way to minimize the interference. We will consider the block allocation in one cluster with the same situation for all three reward schemes. In the pessimistic scheme, as there is only a single positive reward, only the combination of slots which gives that reward will lead to an increasing Q-value. All the other combinations will decrease Q-values and hence will not be selected. In the optimistic scheme, as there are multiple rewards, it is possible that a two-slot combination will always be selected as that too can increase the Q-values. Hence, in the optimistic scheme, we can have multiple slot combinations that can increase the Q-value with a possibility that the Q-value is only maximum for that reward, but not the maximum over all rewards. This is the same case with the balanced scheme, although it will happen less frequently due to a higher difference between the two positive rewards.

## Numerical Results

We illustrate the performance of the algorithm with respect to different rewards in Figure 5.6. We show the evolution of Q-values versus iterations for different reward values. For better illustration, we assume a low value of  $\alpha$  and assume that only one reward is selected till the end of all iterations. In this way, we obtain six evolving Q-values, one for each reward over 1000 iterations. As it can be seen, positive rewards lead to increasing Q-values and negative rewards lead to decreasing Q-values. If the difference of rewards is less, difference between the evolving Q-values is also less, e.g., for reward  $R = 3$  and  $R = 2$ , the Q-values are close, whereas for  $R = 1$  and  $R = 0.5$ , the difference is more. In fact, for  $R = 0.5$ , a similar Q value as  $R = 1$  takes much more iterations, compared to  $R = 2$  and  $R = 3$ . In Table 5.4, comparison between centralized and distributed schemes is presented. This comparison is based on inter and intra-cluster interferences, complexity and number of users per slot. Inter-cluster interference is avoided in fully distributed scheme and minimized in rest of the two. While the intra-cluster interference is avoided in all three schemes. Number of users transmitting in one slot is more than one in centralized and partially distributed schemes while only one user per slot is transmitting in fully distributed case. Centralized scheme has the highest complexity due duplex communication among eNB, controllers and controller, MTCs. Also, it adds to network traffic and network congestion is increased. Partially distributed scheme has higher complexity and communication is between controllers and MTCs. Fully distributed scheme has lower complexity than other two schemes. There is no direct communication among MTCs, controllers and eNB. Only exchange of ACK, NACK signals is done between controllers and associated devices.

In the following section, simulation results are presented to further illustrate the proposed block based Q-learning algorithm.

## 5.4 Results and Discussion

In this section, we present results showing the performance of the proposed distributed algorithm. We do the comparison for the centralized and distributed techniques. We obtain and analyze the convergence time and convergence probability at various SIR thresholds. We further analyze the effect of varying SIR thresholds on the admittance rate of MTCs with the help cumulative success probability.

We are using a grid of  $50\text{m} \times 50\text{m}$ , over which 4 cluster heads and 12 devices are uniformly distributed. Clusters are formed on the basis of spatial distribution. Table 5.5 summarizes the parameters and values used in the simulations.

### 5.4.1 Centralized vs Distributed Implementation

We perform the comparison between centralized and distributed slot allocation and show the superiority of latter in terms of admittance rate. In Figure 5.7, admittance rate for both types of slot allocation at various SIR thresholds is shown. It is evident that more number of MTCs are satisfying the required SIR threshold in distributed allocation. This is due to the repeated reward calculation and resulting slot allocation. Convergence criteria for the block-based Q-learning algorithm is to maximize the number of MTCs qualifying the given SIR threshold. In Figure 5.8, average SIR for both types of allocation for various SIR threshold is shown. Centralized allocation outperforms the distributed allocation, this is due to the less number of MTCs admitted at the same SIR level compared to the distributed. As centralized allocation emphasizes on maximum SIR only rather than maximizing the number of admitted users, which is the case in the distributed case.

Also, distributed allocation performs well in terms of computational complexity and time. We have used block-based Q-learning approach which enables controllers to allocate slots to all of their associated MTCs simultaneously, thus improving the efficiency. In addition to this, there is no involvement of eNB in slot allocation by controllers, which not only reduces the burden on eNB but also significantly reduce the network traffic.

## 5.4.2 Distributed Implementation

Distributed resource allocation is better than centralized in many aspects. In centralized approach, a central controller has access to the global channel and network knowledge, and therefore is capable of making coordination and resource allocation decisions. Therefore, it is responsible of assigning resources to all the devices in the network. Without an efficient and fast infrastructure, centralized allocation is an hard task due to the exchange of inter-cell scheduling information and the large amount of feedback required by the controllers to send all the information. Also, complexity increases exponentially with the increase of devices. The distributed strategy reduces both signaling and feedback requirements as compared to its counter part. Also, distributed schemes are scalable and information exchange and overhead can be adapted according to the size of the network. Hence, in the following, the distributed scheme is further analyzed in detail.

### Reward Schemes

Comparison among various reward schemes is shown in Figures 5.9, 5.10. Convergence probability and convergence time are the parameters used for comparison of various schemes. Performance of algorithm depends upon the reward scheme being used. We can see from the plots that pessimistic reward (R1) performs better than the rest of two schemes. It has highest convergence probability as immediate rewards are biased towards negative values which enables controllers to try different combinations of slot allocation. This increases the possibility of convergence. While for optimistic reward scheme, as immediate rewards are biased towards positive values and there is less probability for controllers to pick up new combinations for slot allocations. Therefore, there are less chances of convergence as controllers will keep on adopting the same slot allocation combination. Most of the time, the algorithm is unable to allocate the slots, the only time it is able to do that is when the initial slot allocation matches the required slot allocation or is very close to the required slot allocation. Thus, when the initial slot allocation is close to the required allocation, the

algorithm converges quickly.

The reason for such a performance and the dependency on reward schemes can be seen by considering the difference between the reward schemes: In the pessimistic scheme, there is only a single positive reward and that is awarded only when all three slots meet the required SIR threshold, meaning that Q-value is only increased in that scenario. In the optimistic scheme, there are multiple positive rewards, it means that even when there is only a single slot that meets the SIR threshold, the reward is still positive, which will increase the Q-value. In the balanced scheme, the Q-value will only increase when a minimum of two slots meet the required SIR threshold. The proposed algorithm selects the block that has the maximum Q-value. In case of pessimistic scheme, that block can only be the one where all slots meet the required threshold. In case of optimistic and balanced schemes, there can be multiple blocks that will lead to a higher Q-value. Thus, it means that the solution given by the optimistic and balanced schemes is not always unique and is dependant on the initialisation.

From above numerical and simulation results, pessimistic reward scheme appears to be the best choice. We apply this reward scheme for the further analysis of our algorithm.

### **Convergence Time**

In Figure 5.11, convergence time is shown with respect to various SIR thresholds. We define the convergence time as the time taken by the controllers to learn with repeated data transmission and resulting  $Q$  values, for the selection of best block and best time slots for their data transmission. Convergence is declared when there is no change in block selection and in turn slot assignment. We calculate the convergence time obtained over 1000 iterations and average the results. Different values of SIR varying from 2 to 10 in a step-size of 2 are used to study the change in convergence time. Convergence time varies from 2 to 9 slots. The convergence time increases as the SIR threshold is increased. This is because, with increasing threshold, there are less configurations of MTCDs that can meet the required threshold. It was expected as it is difficult for all the MTCDs to meet the required SIR threshold and it takes more time by controllers to learn with experience that how to pick up

the best block.

### **Convergence Probability**

Figure 5.12 shows convergence probability for various SIR thresholds. Convergence probability is defined as ratio of number of times simulation converge and the total number of iterations. SIR is varied from 2 to 10 in a step size of 2. A decreasing trend in convergence probability can be observed between 0.5 and 0.9. We can see that when SIR threshold is increased, then convergence probability is decreased. This is because with the increase of SIR threshold, less configurations of MTCDs can fulfill the required thresholds as also seen in Figure 5.11. Hence, the simulation converges less number of times as the SIR threshold increases indicating that it is difficult for all the MTCDs to get desired threshold simultaneously.

### **Admittance Rate**

In Figure 5.13, cumulative distribution function is used to study the effect of SIR threshold on number of MTCDs meeting the required SIR level. It shows only the number of successful MTCDs at each threshold level provided all of them meet the convergence criteria. For example at  $\gamma = 10$ , 50 percent of times, all of the devices meet the required SIR criteria (that is all the MTCDs will get the SIR greater than SIR threshold). We can see that as the SIR level increases, convergence probability decreases and not all the devices meet the required threshold. As SIR threshold is decreased, increase in number of satisfying MTCDs is observed. When the number of MTCDs is 8 and the admittance rate is 0.5, it means that only 4 MTCDs are admitted. For  $\gamma = 2$ , all the MTCDs are admitted as the threshold is low enough. This can also be seen by Figures 5.11, 5.12, where at  $\gamma = 2$ , convergence time is minimum and convergence probability is maximum. For  $\gamma = 4$  to 8, the admittance rate starts decreasing from 10 MTCDs and goes to 0.5 for 12 MTCDs. In case of  $\gamma = 10$ , the admittance rate is the lowest. A decreasing trend of admittance with respect to SIR threshold can be seen, which shows that with an increase of SIR threshold, lower number of

MTCDS will be admitted.

Figure 5.14 shows the percentage of MTCDS satisfying the given SIR threshold criteria. Three SIR levels 2, 6 and 10 are used do the analysis. We can see that for  $\gamma = 2$ , 90 percent of times all MTCDS reach the required level while for  $\gamma = 6$ , this number reduces to 75 percent and for  $\gamma = 10$  to 30 percent.

## 5.5 Chapter Summary

We carried out slot allocation in random access network of MTCDS in a centralized/distributed manner and presented a strategy for the selection of time slots by MTCDS controllers. We showed the superiority of distributed assignment over centralized in terms of complexity and average SIR. We have used block-based Q-learning for the resource allocation for MTCDS in a distributed manner. We explored the relationship between SIR threshold and convergence probability. It is observed that convergence probability decreases with the increase in SIR threshold. We further showed that it takes longer to converge at higher values of SIR threshold.

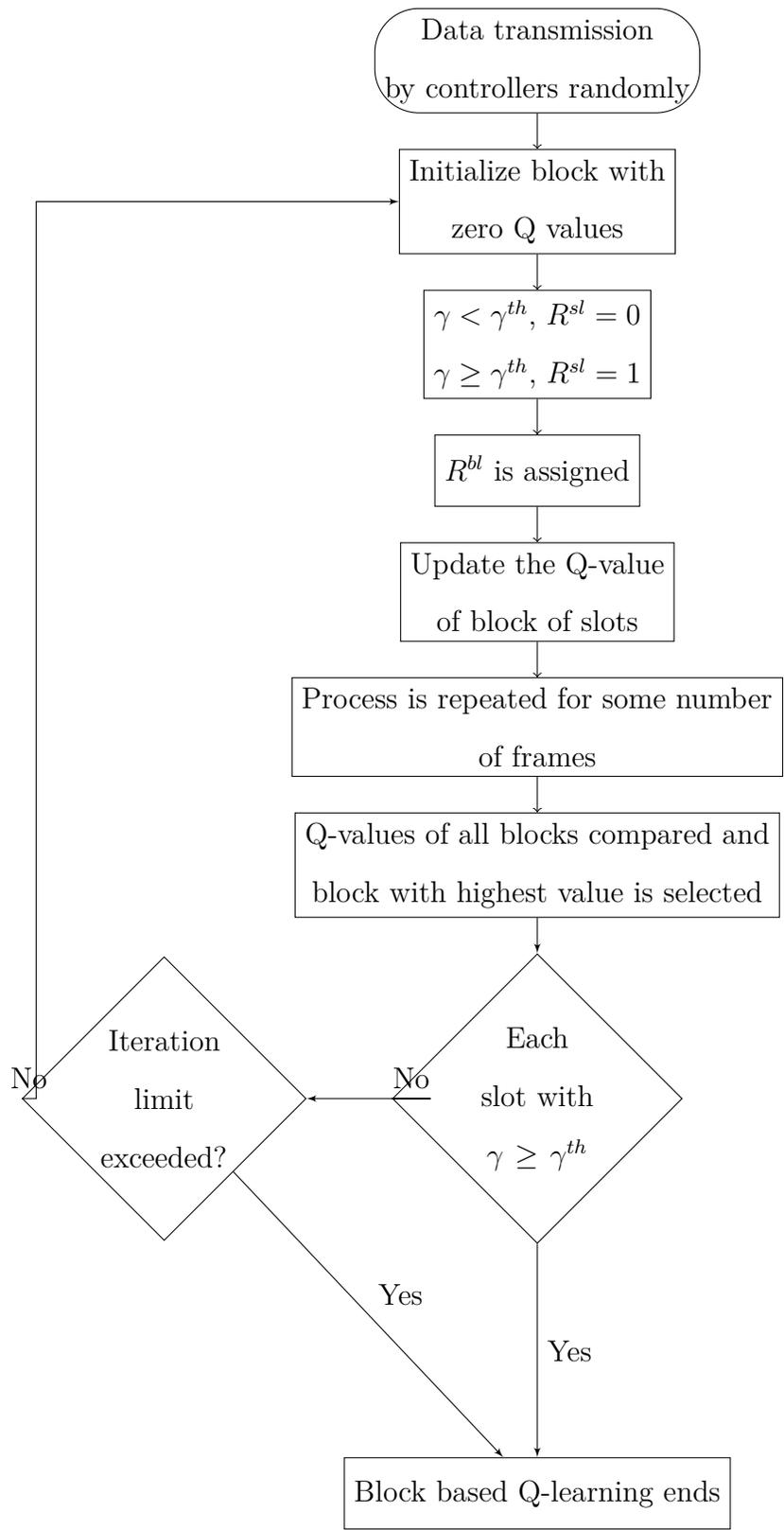


Figure 5.5: Block Q-learning Flow Chart

Table 5.3: Comparison of Reward Schemes

Block selection	Pessimistic Reward		Optimistic Reward		Balanced Reward		Time
	Block	Q-value	Block	Q-value	Block	Q-value	
$U_1U_2U_3$	✓	-0.5	✓	1	✓	0.25	t=1
$U_1U_3U_2$		0		0		0	
$U_2U_1U_3$		0		0		0	
$U_2U_3U_1$		0		0		0	
$U_3U_1U_2$		0		0		0	
$U_3U_2U_1$		0		0		0	
$U_1U_2U_3$		-0.5	✓	1.5	✓	-.125	t=2
$U_1U_3U_2$	✓	-1				0	
$U_2U_1U_3$		0				0	
$U_2U_3U_1$		0				0	
$U_3U_1U_2$		0				0	
$U_3U_2U_1$		0				0	
$U_1U_2U_3$		-0.5	✓	0.5		-.125	t=3
$U_1U_3U_2$		-1		0	✓	0.25	
$U_2U_1U_3$	✓	1		0		0	
$U_2U_3U_1$		0		0		0	
$U_3U_1U_2$		0		0		0	
$U_3U_2U_1$		0		0		0	

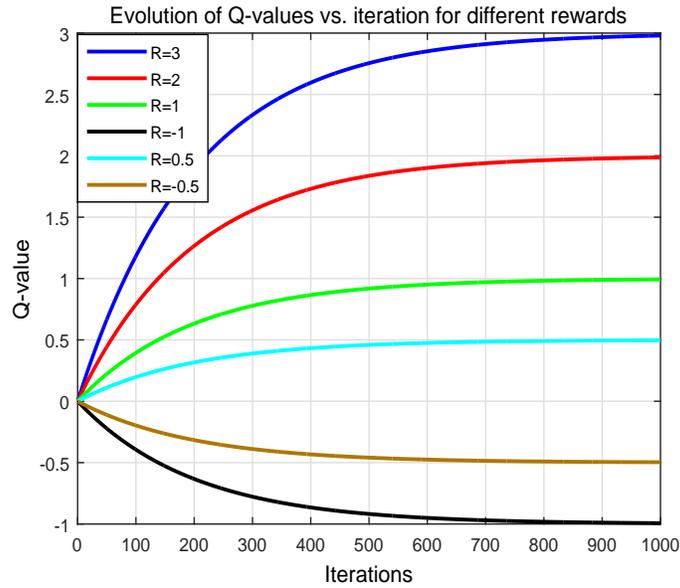


Figure 5.6: Evolution of Q-values vs Iteration for Various Rewards

Table 5.4: Comparison between Centralized and Distributed Schemes

Attributes	Centralized	Fully Distributed [18]	Partially Distributed
Inter-cluster Int.	✓	×	✓
Intra-cluster Int.	×	×	×
Users per slot	> 1	1	> 1
Communication	eNb ↔ controller ↔ MTCD	ACK, NACK	controllers ↔ MTCDs
Complexity	high	low	high

Table 5.5: Notations and Values

Symbol	Definition
$K = 4$	Number of clusters/MTCD controllers
$D = 3$	Number of devices in each cluster
$T = 3$	Number of slots in each frame
$U_j$	Set of MTCDs in $j^{th}$ cluster
$\gamma^{th} = 2, 4, \dots, 10$	SIR Threshold
$P_r = 1 \text{ mW}$	Received power

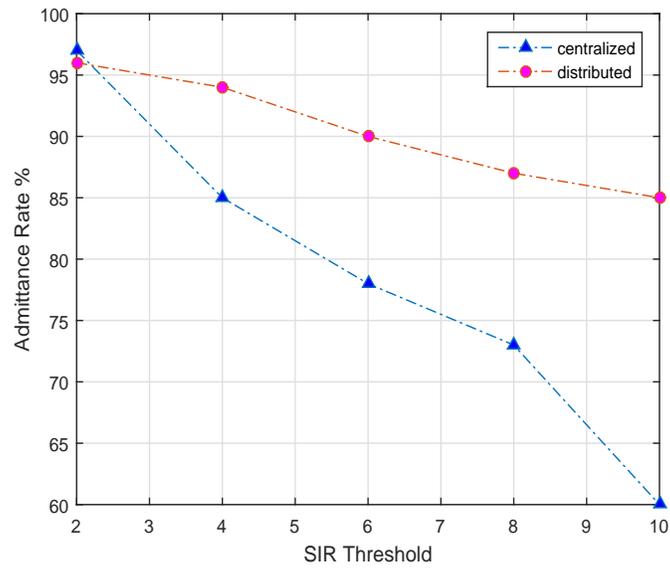


Figure 5.7: Admittance Rate for Centralized and Distributed Allocation

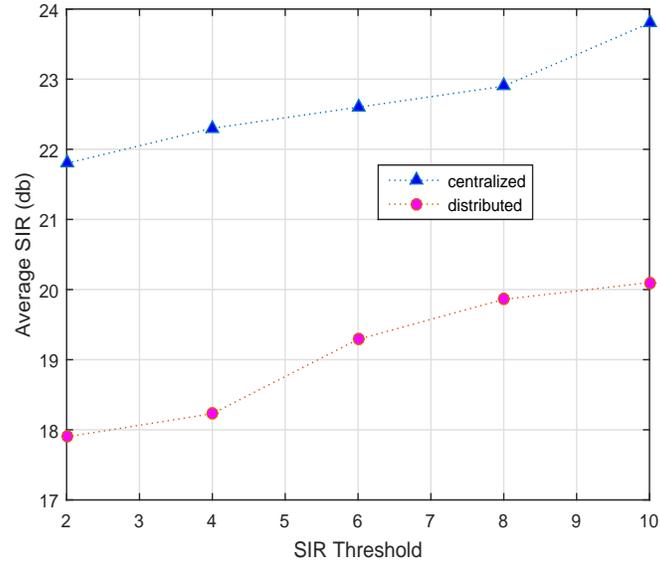


Figure 5.8: Average SIR for Centralized and Distributed Allocation

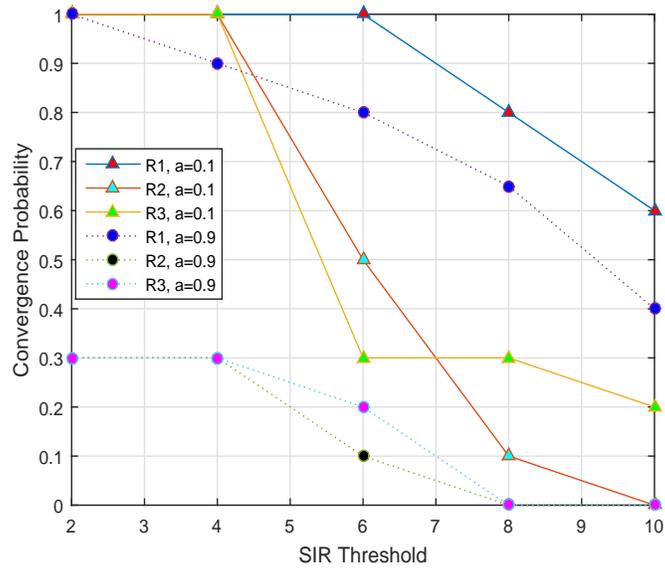


Figure 5.9: Convergence Probability at Various Reward Schemes; R1:Pessimistic, R2: Optimistic, R3:Balanced

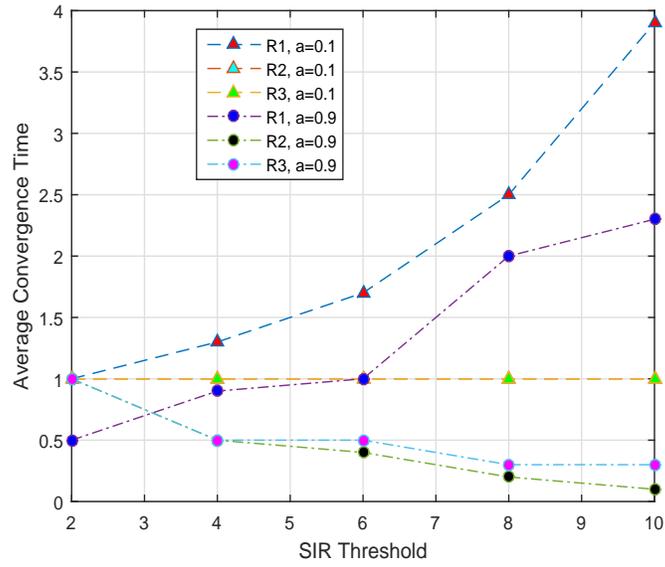


Figure 5.10: Convergence Time at Various Reward Schemes; R1:Pessimistic, R2: Optimistic, R3:Balanced

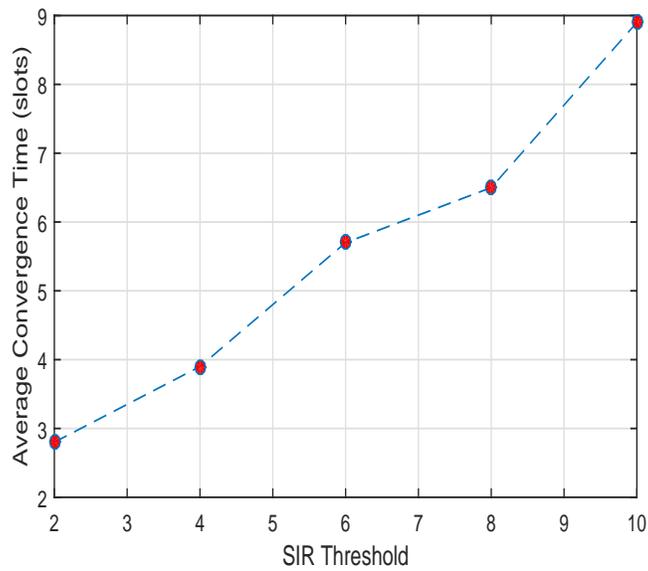


Figure 5.11: Convergence Time at Various SIR Thresholds

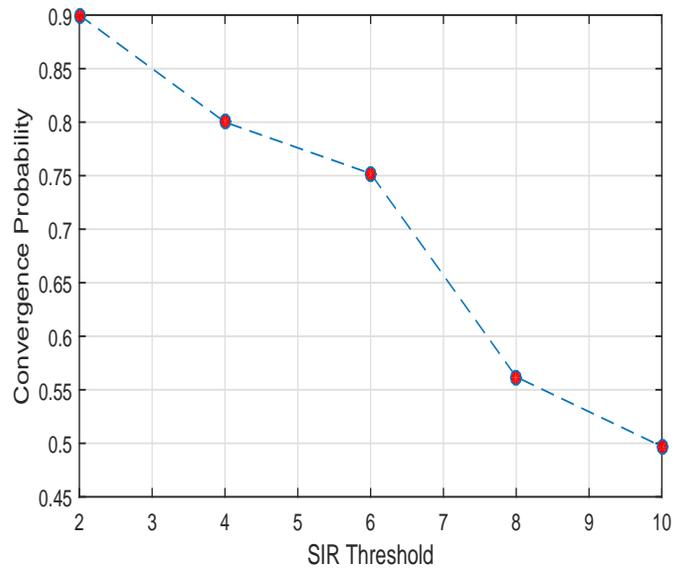


Figure 5.12: Convergence Probability at Various SIR Thresholds

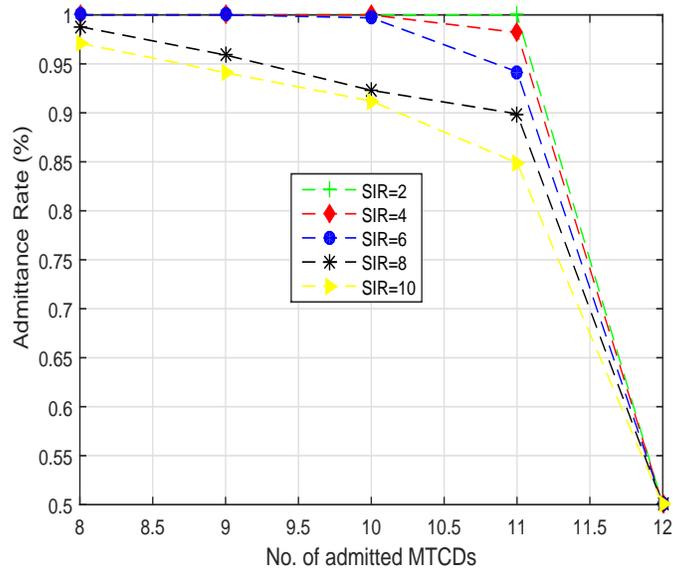


Figure 5.13: Successful MTCs at Various SIR Thresholds

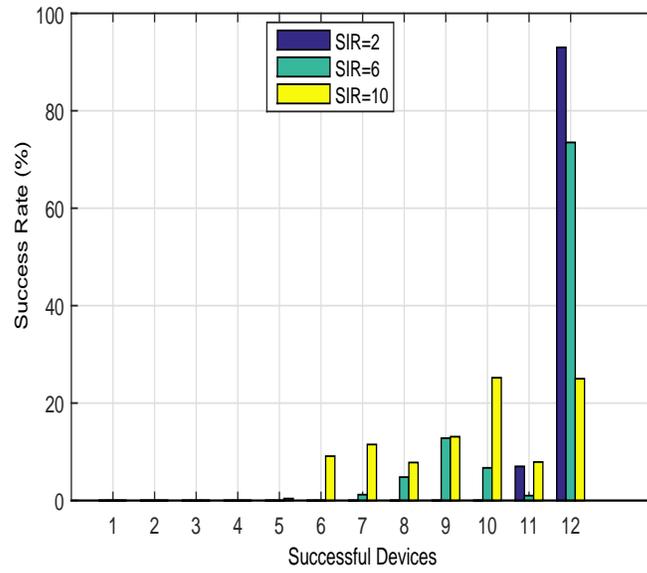


Figure 5.14: Successful MTCs at SIR Threshold 2, 6, 8

# Chapter 6

## Conclusions and Future work

This dissertation mainly focused on resource allocation of MTCs in an M2M network. MTCs are grouped into cluster to accommodate massive random access, as clustering and group based MAC has been suggested by many researchers and seem efficient and applicable. Each cluster is managed by a cluster head and all cluster heads or MTC controllers will receive aggregated data from eNB and distribute to its associated devices.

### 6.1 Conclusion

We tried to maximize the number of admitted MTC controllers into the network with least interference caused to conventional human users. We formulated the problem as a mixed-integer non-linear problem and solved it using meshed adaptive direct search algorithm. Afterwards we did scheduling of MTCs within a cluster. At first, we formulated the slot selection problem as an optimization problem. Then we presented a solution using the Q-learning algorithm to select conflict-free slot assignment in a random access network with MTC controllers. In continuation to the above, we presented a block based Q-learning algorithm for the scheduling of MTCs in distributive manner. At first centralized slot assignment is done and an algorithm is proposed for minimizing the inter-cluster interference. Then we used Q-learning algorithm to assign slots in a distributed manner and comparison

is made between the two schemes. Effort is made to avoid or minimize the intra-cluster and inter-cluster interference and to ensure at least the minimum SIR requirements of MTCDS in a cluster.

## 6.2 Future Work

- Formation of M2M clusters on the basis of energy efficiency and QoS is still not much explored topic. Due to unique properties of MTCDS, we aim to investigate and implement QoS based clustering. To schedule mixed and individual traffic without affecting human users is a task. We intend to form mixed traffic and individual traffic clusters. Scheduling for both types of clusters is performed and comparison should be done for analyzing the validity and usefulness of a specific criteria. However, it is challenging to make/remake groups/clusters in real time with specific QoS requirements.
- We intend to introduce the concept of hybrid MAC protocol for MTCDS's channel access. As in the mixed traffic case, various data rates and priority devices have to be accommodated without deteriorating their QoS, scheduling is a challenge. MTCDS belonging to different classes will require different channel/slot/access grant interval time (AGTI) for fulfilling their required data rate demands. We plan to use Z-MAC for the scheduling of mixed traffic within a cluster. Z-MAC combines the strengths of TDMA and CSMA and, we want to benefit from its adaptability to the level of contention in the network. We plan to make two classes, for instance, one with periodic access but lesser number of required slots and the second with priority access but not so frequent with larger slots requirement. Z-MAC will accommodate both types of devices with improved network throughput.
- We aim to use frequency hopping in a clustered M2M network to avoid the signalling overhead in energy limited MTCDS. The key challenge is the design of hopping sequences to guarantee interference-free communication. Existence of large number of

MTCs will result into longer hopping sequence which further results into congestion due to longer waiting time. We aim to design an distributed and adaptive multichannel hopping protocol which can efficiently accommodate varying MTC traffic with less delay.

- In this thesis, we used Q-learning with in an distributive shared slot TDMA environment. For the future work, we aim to use Q-learning with in an distributive shared slot OFCDM environment, in which, controllers will learn with experience which code/spreading is preferred for them in terms of lowest interference and best channel gain. MTCs will be able to choose between time domain spreading and frequency domain spreading according to the amount of traffic in a network.
- For admitting MTC controllers using non-linear programming, simple case of existence of only three human/primary users and ten MTCs was considered in our present work. We aim to extend it for larger number of both types of users with different distributions. We also aim to examine the effects of all three performance metrics with different weights.

# Bibliography

- [1] J. Cheng and C. Lee, “A Novel Random Access for Fixed-Location Machine-to-Machine Communications in OFDMA Based Systems,” *IEEE Communication Letters*, vol. 16, pp. 1428 – 1431, September 2012.
- [2] J. Cheng and C. Lee, “Prioritized Random Access with Dynamic Access Barring for RAN Overload in 3GPP LTE-A Networks,” *IEEE GLOBECOM Workshops (GC Workshops)*, pp. 368 – 372, 2011.
- [3] C. Tu and C. Ho, “Energy-Efficient Algorithms and Evaluations for Massive Access Management in Cellular Based Machine To Machine Communications,” *IEEE Vehicular Technology Conference*, pp. 1 – 5, Fall 2011.
- [4] E. Dahlman and G. Mildh, “5G Wireless Access: Requirements and Realization,” *IEEE Communications Magazine*, pp. 42– 47, December 2014.
- [5] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, and J. Wang, “Large-Scale Measurement and Characterization of Cellular Machine-to-Machine Traffic ,” *IEEE/ACM Transactions on Networking*, vol. 21, pp. 1960– 1973, December 2013.
- [6] R. K. Lam and K.-C. Chen, “Congestion Control for M2M Traffic with Heterogeneous Throughput Demands,” *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1452– 1457, April 2013.

- [7] S. Jalali, “M2M Solutions Ū Design Challenges and Considerations ,” *IEEE Recent Advances in Intelligent Computational Systems (RAICS)*, pp. 2010– 2014, December 2013.
- [8] A. Lo and M. Jacobsson, “A Cellular-Centric Service Architecture for Machine-to-Machine (M2M) Communications,” *IEEE Wireless Communications*, vol. 20, pp. 143–151, October 2013.
- [9] I. Abdalla and S. Venkatesan, “Scalable Addressing of M2M Terminals in 4G Cellular Wireless Networks,” *Wireless Telecommunications Symposium*, pp. 1– 6, April 2013.
- [10] W. Yuan and W. Cheng, “Variable-Width Channel Allocation for Access Points: A Game-Theoretic Perspective,” *IEEE Transactions On Mobile Computings*, vol. 12, pp. 1428– 1442, July 2013.
- [11] M. A. Alsheikh, D. Niyato, and H.-P. T. Lin, “Machine Learning in Wireless Sensor Networks: Algorithms, Strategies, and Applications,” *IEEE Communication Surveys and Tutorials*, vol. 16, pp. 1996– 2018, November 2014.
- [12] A. Aijaz and A. Aghvami, “On Radio Resource Allocation in LTE Networks with Machine-to-Machine Communications,” *IEEE Vehicular Technology Conference*, pp. 1– 5, June 2013.
- [13] A. Bartoli and M. Dohler, “Secure Lossless Aggregation Over Fading and Shadowing Channels for Smart Grid M2M Networks,” *IEEE Transactions on Smart Grid*, vol. 2, pp. 571– 572, DECEMBER 2011.
- [14] C. Ho and C. Huang, “Energy-Saving Massive Access Control and Resource Allocation Schemes for M2M Communications in OFDMA Cellular Networks,” *IEEE Wireless Communications Letters*, vol. 1, pp. 209 – 212, June 2012.

- [15] A. Lo and Y. W. Law, “Enhanced LTE-Advanced Random-Access Mechanism for Massive Machine-to-Machine (M2M) Communications,” *IEEE Transaction on Wireless Communications*(*check*), vol. 11, pp. 27 – 32, January 2012.
- [16] F. Hussain, A. Anpalagan, and R. Vannithamby, “Medium Access Control Techniques in M2M Communication: Survey and Critical Review,” *Wiley Transactions on Emerging Telecommunications Technologies*, September 2014.
- [17] F. Hussain, A. Anpalagan, A. S. Khwaja, and M. Naeem, “Slot Allocation with Interference Minimization using Block based Q-Learning in Clustered M2M Networks (submitted),” *Canadian Journal of Electrical and Computer Engineering (CJECE)*, July 2015.
- [18] F. Hussain, A. Anpalagan, A. S. Khwaja, and M. Naeem, “Resource Allocation and Congestion Control in Clustered M2M Communication using Q-Learning (Revised),” *Wiley Transactions on Emerging Telecommunications Technologies*, 2015.
- [19] F. Hussain, A. Anpalagan, A. S. Khwaja, and M. Naeem, “Multi-Objective Resource Allocation in M2M Communication (Revised),” *IET Communications*, March 2015.
- [20] F. Hussain, A. Anpalagan, A. S. Khwaja, and M. Naeem, “Multi-Objective MTC Device Controller Resource Optimization in M2M Communication,” *Queens Biennial Symposium on Communication(QBSC)*, pp. 184–188, June 2014.
- [21] A.-S. Tien Nguyen and E.-N. Huh, “ A dynamic ID-based authentication scheme ,” *International Conference on Networked Computing and Advanced Information Management (NCM)*, pp. 248–253, 2010.
- [22] S. Pill and S. J. Kang, “A Study on the Performance Evaluation of container tracking device based on M2M ,” *International Conference on Advanced Communication Technology*, pp. 67– 72, February 2014.

- [23] N. Walravens and P. Ballon, “Platform Business Models for Smart Cities: From Control and Value to Governance and Public Value,” *ieecm*, pp. 76– 83, June 2013.
- [24] C. Mulligan and M. Olsson, “Architectural Implications of Smart City Business Models: An Evolutionary Perspective,” *ieecm*, pp. 84– 89, June 2013.
- [25] A. Cimmino and T. Pecorella, “The role of small cell technology in future Smart City applications,” *IEEE Transactions on Emerging Telecommunications Technologies*, pp. 11– 20, November 2014.
- [26] A. Morales and T. Robles, “A Reference Mobile System for Extending Content in Dynamic Domestic Environments,” *IEEE International Conference on Consumer Electronics*, pp. 404– 405, January 2014.
- [27] D. Bodson, “A reference mobile system for extending content in dynamic domestic environments ,” *IEEE Vehicular Technology Magazine*, vol. 9, pp. 22– 27, March 2014.
- [28] D. Fu and J. Solinas, “IKE and IKEv2 Authentication Using the Elliptic Curve Digital Signature Algorithm (ECDSA),” *Internet Engineering Task Force*, January 2007.
- [29] N. Ahmed, “M2M-Based Service Coverage for Mobile Users in Post-Emergency Environments ,” *IEEE Transaction on Vehicular Technology*, vol. pp, p. 1, January 2014.
- [30] K. E. Seong and S. J. Kang, “Self M2M based wearable watch platform for collecting personal activity in real-time ,” *International Conference on Big Data and Smart Computing*, pp. 286– 290, January 2014.
- [31] M. Chen and J. Wan, “A Survey of Recent Developments in Home M2M Networks ,” *IEEE Communications Surveys and Tutorials*, vol. 16, pp. 98– 114, February 2014.
- [32] Z. Fan and P. Kulkarni, “M2M Communications for E-Health and Smart Grid : An Industry and Standard Perspective,” *IEEE Wirelss Communications*, vol. 21, p. 62 69, March 2014.

- [33] S. Lien and K. Chen, "Toward Ubiquitous Massive Accesses in 3GPP Machine-to-Machine Communications," *IEEE Communications Magazine*, vol. 49, pp. 66 – 74, April 2011.
- [34] M. Sallinen, "Wireless M2M Communication in Professional Applications," *International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 571– 572, November 2010.
- [35] G. Roch, "Application Architectures for Machine to Machine Communications: Research Agenda vs. State-Of-The Art," *International Conference on Broadband Communications and Biomedical Applications (IB2Com)*, pp. 1– 5, November 2011.
- [36] P. Stenumgaard and J. Chilo, "Challenges and Conditions for Wireless Machine-to-Machine Communications in Industrial Environments," *ieecm*, pp. 191– 196, June 2013.
- [37] D. Katusic and M. Weber, "Market, Standardization, and Regulation Development in Machine-to-Machine Communications ," *International Conference on Software, Telecommunications and Computer Networks*, pp. 1 – 7, September 2012.
- [38] ETSI-TR-102-935-V2.1.1, "Machine-to-Machine communications (M2M); Applicability of M2M architecture to Smart Grid Networks; Impact of Smart Grids on M2M platform ," *European Telecommunications Standards Institute*, September 2012.
- [39] ETSI-TS-102-690-V2.1.1, "Machine-to-Machine communications (M2M); Functional architecture ," *European Telecommunications Standards Institute*, October 2013.
- [40] M. Sneps-Sneppé and D. Namiot, "About M2M standards and their possible extension ," *Baltic Congress on Future Internet Communication*, pp. 187 – 193, April 2012.
- [41] F. J. Lin and E. Cerritos, "A Feasibility Study on Developing IoT/M2M Applications over ETSI M2M Architecture ," *International Conference on Parallel and Distributed System*, pp. 558 – 563, December 2013.

- [42] 3GPP-TR-37.869-V1.0.0, “Study on Enhancements to Machine Type Communications (MTC) and other Mobile Data Applications;,” *3rd Generation Partnership Project*, August 2013.
- [43] M. Fuertes and F. Casado, “Evaluation of Latency-Aware Scheduling Techniques for M2M Traffic over LTE,” *European Signal Processing Conference (EUSIPCO)*, pp. 989 – 993, 2012.
- [44] K. Yano and K. Murakami, “Dynamic and Reconfigurable M2M Wireless Network with QoE-based Control,” *Future Network and Mobile Summit*, pp. 1– 9, July 2013.
- [45] A. Alexiou and A. Gotsis, “Evolution of Packet Scheduling for Machine-Type Communications over LTE: Algorithmic Design and Performance Analysis,” *IEEE Global Telecommunications workshop*, pp. 1745 – 1750, 2012.
- [46] P. Forero and A. Cano, “Distributed Clustering using Wireless Sensor Networks,” *IEEE Transaction on Selected Topics in Signal Processing*, vol. 50, pp. 707– 724, August 2011.
- [47] T.-C. Juan and S.-E. Wei, “Data-Centric Clustering for Data Gathering in Machine-to-Machine Wireless Networks,” *IEEE Conference on Communications*, pp. 89– 94, June 2013.
- [48] Y. Liu and C. Yuen, “Design of A Scalable Hybrid MAC Protocol for Heterogeneous M2M Networks ,” *IEEE Journal of Internet of Things*, vol. pp, March 2014.
- [49] R. Ma and H. Chen, “Smart Grid Communication: Its Challenges and Opportunities,” *IEEE Transaction On Smart Grid*, vol. 4, pp. 36– 46, March 2013.
- [50] A. Sikora and A. Yunitasari, “GPRS and UMTS services for ultra low energy M2M-communication ,” *Internation Conference of Intelligent Data Acquisition and Advanced Computing Systems*, vol. 1, pp. 494– 498, September 2013.

- [51] F. Pauls and S. Krone, “Evaluation of Efficient Modes of Operation of GSM/GPRS Modules for M2M Communications ,” *IEEE VTC*, vol. 1, pp. 1– 6, September 2013.
- [52] A. Lioumpas, A. Alexiou, and A. G. Gotsis, “M2M Scheduling over LTE; Challenges and New Perspectives,” *IEEE Vehicular Technology Magazine*, vol. 7, pp. 34 – 39, August 2012.
- [53] e. a. Francisco Vazquez-Gallego, “Performance Evaluation of Frame Slotted-ALOHA with Successive Interference Cancellation in Machine-to-Machine Networks,” *IEEE European Wireless Conference*, pp. 403– 408, May 2014.
- [54] M. Hasan and E. Hossain, “Random Access for Machine-to-Machine Communication in LTE-Advanced Networks: Issues and Approaches,” *ieecm*, pp. 90– 97, June 2013.
- [55] L. Grieco and G. Boggia, “Downlink Packet Scheduling in LTE Cellular Networks: Key Design Issues and a Survey,” *IEEE Communications Surveys and Tutorials*, pp. 1 – 23, 2012.
- [56] R. Bendlin and J. Nossek, “Two-Phase Scheduling and Leakage-Based Precoding in Wireless Cellular Networks,” *IEEE Transaction On Wireless Communications*, vol. 12, pp. 3148– 3157, July 2013.
- [57] M. Bennis and W. Saad, “When Cellular Meets WiFi in Wireless Small Cell Networks,” *ieecm*, pp. 48– 54, June 2013.
- [58] O. Delgado and B. Jaumard, “Scheduling and Resource Allocation for Multiclass Services in LTE Uplink systems,” *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pp. 355 – 360, 2010.
- [59] A. Alexiou and A. Gotsis, “Analytical modelling and performance evaluation of realistic time-controlled M2M scheduling over LTE cellular networks,” *IEEE Transactions on Emerging Telecommunications Technologies*, pp. 378– 388, March 2013.

- [60] Y. Zaki and T. Weerawardane, “Multi-QoS-aware Fair Scheduling for LTE,” *IEEE Vehicular Technology Conference*, pp. 1– 5, 2011.
- [61] K. Zheng and F. Hu, “Radio Resource Allocation in LTE-Advanced Cellular Networks with M2M Communications,” *IEEE Communications Magazine*, vol. 50, no. 7, pp. 184 – 192, 2012.
- [62] K. D. Chia Yu, “ Resource Sharing Optimization for Device-to-Device Communication Underlying Cellular Networks ,” *IEEE Transactions on Wireless Communication*, vol. 10, pp. 2752 – 2763, 2013.
- [63] S. Tsong and C. Hsiang, “Self-Adaptive Persistent Contention Scheme for Scheduling Based Machinetype Communications in LTE System,” *International Conference on Selected Topics in Mobile and Wireless Networking (ICOST)*, pp. 77– 82, 2012.
- [64] A. Lioumpas and A. Alexiou, “Uplink Scheduling for Machine to Machine Communications in LTE-based Cellular Systems,” *IEE GLOBECOM Workshops (GC Wkshps)*, pp. 353–357, May 2011.
- [65] L. Karim and A. Anpalagan, “Fault tolerant, energy efficient and secure clustering scheme for mobile machine-to-machine communications,” *IEEE Transactions on Emerging Telecommunications Technologies*, pp. 1– 18, 2014.
- [66] L. Shao and K. Cheng, “§Massive Access Management for QoS Guarantees in 3GPP Machine-to-Machine Communications,Ŧ,” *IEEE Global Telecommunications workshop*, vol. 15, no. 3, pp. 311 – 313, 2011.
- [67] A. Laya and J. Alonso-Zarate, “Is the Random Access Channel of LTE and LTE-A Suitable for M2M Communications? A Survey of Alternatives,” *IEEE Communications Surveys and Tutorials*, vol. 16, pp. 4– 16, February 2014.

- [68] M. Gerasimenko and S. Andreev, “Impact of machine-type communications on energy and delay performance of random access channel in LTE-advanced,” *IEEE Transactions on Emerging Telecommunications Technologies*, pp. 366– 377, April 2013.
- [69] K. Lee and S. Kim, “Throughput Comparison of Random Access Methods for M2M Service over LTE Networks,” *IEEE GLOBECOM Workshops (GC Wkshps)*, pp. 373 – 377, 2011.
- [70] K. Zhou and N. Nikaein, “Contention based access for machine-type communications over LTE,” *IEEE Vehicular Technology Conference*, no. 9, pp. 1 – 5, 2012.
- [71] G. Wang and X. Zhong, “§An Adaptive Medium Access Control Mechanism for Cellular Based Machine to Machine (M2M) Communication,” *EEE International Conference on Wireless Information Technology and Systems (ICWITS)*, pp. 1 – 4, 2010.
- [72] R. C. Paiva and R. Vieira, “Random access capacity evaluation with synchronized MTC users over wireless networks,” *IEEE Vehicular Technology Conference*, pp. 1– 5, May 2011.
- [73] J.-B. Seo and V. Leung, “Design and Analysis of Backoff Algorithms for Random Access Channels in UMTS-LTE and IEEE 802.16 Systems,” *ieevtc*, vol. 60, pp. 3975– 3989, October 2013.
- [74] Z. Jiang and X. Zhong, “§Fast Retrial and Dynamic Access Control Algorithm for LTE-Advanced Based M2M Network,” *Advanced International Conference on Telecommunications (AICT)*, pp. 24 – 28, 2012.
- [75] e. a. Francisco Vazquez-Gallego, “Performance Evaluation of Reservation Frame Slotted Aloha for Data Connection M2M Networks,” *IEEE European Wireless Conference*, pp. 633– 638, May 2014.

- [76] G. Elkheir and A. Alexiou, “Energy Efficient Cooperative Scheduling based on Sleep-Wake Mechanisms,” *IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, pp. 238– 242, April 2012.
- [77] S. Andreev and O. Galinina, “Energy-Efficient Client Relay Scheme for Machine-to-Machine Communication,” *IEEE Global Telecommunications Conference*, no. 9, pp. 1 – 5, 2011.
- [78] C. Ho and C. Huang, “Analysis of Delay-Energy Tradeoff and Energy Minimization Schemes for Group-based Machine-to-Machine Communications in OFMDA Cellular Networks,” *IEEE Vehicular Technology Conference (VTC Fall)*, pp. 1– 5, 2012.
- [79] R. Yu and Y. Zhang, “Distributed Rate and Admission Control in Home M2M networks: A Non-cooperative Game Approach,” *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 196– 200, 2011.
- [80] K. Jun, “Call Admission Control Satisfying Delay Constraint for Machine-to-Machine Communications in LTE-Advanced,” *Communications in Computer and Information Scienc*, Springer, vol. 350, pp. 48– 55, 2012.
- [81] S. Lien and T. Liao, “Cooperative Access Class Barring for Machine-to-Machine Communications,” *IEEE Transaction on Wireless Communications*, vol. 11, pp. 27 – 32, January 2012.
- [82] T. Ogawa and S. Kotabe, “Fair Congestion Control Method for Terminal Groups with Wireless Random Access in M2M Network,” *International Symposium on Communications and Information Technologies (ISCIT)*, pp. 1069– 1074, 2012.
- [83] N. Pratas and H. Thomsen, “Code-Expanded Random Access for Machine-Type-Communications,” *IEEE Global Telecommunications workshop*, pp. 1806– 1811, 2012.

- [84] U. Phuyal and R. Vannithamby, "Controlling access overload and signaling congestion in M2M networks," *Asilomar Conference on Signals, Systems and Computers (ASIL-MOR)*, pp. 591– 595, 2012.
- [85] B. Yang and G. Zhu, "M2M access performance in LTE-A system," *IEEE Transactions on Emerging Telecommunications Technologies*, pp. 1– 8, November 2013.
- [86] K. Lee and M. Reisslein, "Handling Randomness of Multi-Class Random Access Loads in LTE-Advanced Network Supporting Small Data Applications," *IEEE International Workshop on Heterogeneous, Multi-Hop, Wireless and Mobile Networks*, pp. 436– 440, 2012.
- [87] Y. Chai and P. Ren, "Partial Time-Frequency Resource Allocation for Device-to-Device Communications Underlying Cellular Networks," *icc*, pp. 4648– 4652, June 2013.
- [88] T. Jiang and X. Tan, "Evolutionary game based access class barring for machine-to-machine communications ," *International Conference on Advanced Communication Technology*, vol. 43, pp. 832– 835, February 2014.
- [89] A. Larmo and R. Susitaival, "RAN Overload control for Machine-Type-Communications in LTE," *IEEE Global Telecommunication Workshop*, 2012.
- [90] A. Larmo and T. Tirronen, "Machine-to-machine communication with long-term evolution with reduced device energy consumption," *IEEE Transactions on Emerging Telecommunications Technologies*, pp. 413– 426, April 2013.
- [91] V. Mancuso and A. Asadi, "Code-expanded radio access protocol for machine-to-machine communications," *IEEE Transactions on Emerging Telecommunications Technologies*, pp. 355– 365, May 2013.
- [92] C. Seunghyun and L. Wonbo, "Automatic configuration of random access channel parameters in LTE systems," *IEEE IFIP Wireless Days*, pp. 1 – 6, 2011.

- [93] S. Lu and S. Sheu, "Load Analysis for MTC Devices in Idle Mode or Detached State," *International Computer Symposium (ICS)*, pp. 424– 428, 2010.
- [94] R. Zhang and L. Song, "Distributed Resource Allocation for Device-to-Device Communications Underlying Cellular Networks," *icc*, pp. 482– 486, June 2013.
- [95] T. Biermann and L. Scalia, "How Backhaul Networks Influence the Feasibility of Coordinated Multipoint in Cellular Networks," *IEEE Communications Magazine*, pp. 172– 180, August 2013.
- [96] R. R. Tyagi and K.-D. Lee, "Impact of Retransmission Limit on Preamble Contention in LTE-Advanced Network," *IEEE Systems Journal*, pp. 1– 14, 2013.
- [97] e. a. Orazio Briante, "Duty-Cycle Optimization for Machine-to-Machine Area Networks Based on Frame Slotted-ALOHA with Energy Harvesting Capabilities," *IEEE European Wireless Conference*, pp. 1– 6, May 2014.
- [98] B. Krishnamachari and G. Lu, "An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Wireless Sensor Networks," *International Symposium on Parallel and Distributed Processing (IPDSP)*, pp. 224– 231, April 2004.
- [99] A. B. Nacef and S. M. Senouci, "A Cooperative Low Power Mac Protocol for Wireless Sensor Networks," *IEEE Conference on Communications*, pp. 1– 6, July 2011.
- [100] L. Zheng and L. Cai, "Efficient Multi-recipient Message Aggregation for Short Message Delivery in M2M Networks," *IEEE Transactions on Smart Grid*, 2011.
- [101] J. Matamoros and C. Haro, "Data Aggregation Schemes for Machine-to-Machine Gateways: Interplay with MAC Protocols," *Future Network and Mobile Summit (FutureNetw)*, July 2012.

- [102] Y.-D. Tsai, C.-Y. Song, and H.-Y. Hsieh, “Joint Optimization of Clustering and Scheduling for Machine-to-Machine Communications in Cellular Wireless Networks,” *IEEE Vehicular Technology Conference*, pp. 1– 5, May 2015.
- [103] D. Mur and A. Saavedra, “Resource Allocation for Device-to-Device Communications Overlaying Two-Way Cellular Networks,” *IEEE Transaction On Wireless Communications*, vol. 12, pp. 3611– 3621, July 2013.
- [104] e. a. Tatjana Predojević, “Energy Evaluation of a Cooperative and Duty-Cycled ARQ Scheme for Machine-to-Machine Communications with Shadowed Links,” *International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1640– 1644, September 2013.
- [105] S.-E. Wei and H.-Y. Hsieh, “Joint Optimization of Cluster Formation and Power Control for Interference-Limited Machine-to-Machine Communications,” *IEEE globecom*, pp. 5512 – 5518, December 2012.
- [106] S.-E. Wei and H.-Y. Hsieh, “Enabling dense machine-to-machine communications through interference-controlled clustering ,” *Wireless Communications and Mobile Computing Conference*, pp. 774 – 779, August 2012.
- [107] S.-E. Wei and M. Ho, “Random Access Design for Clustered Wireless Machine to Machine Networks,” *First International Black Sea Conference on Communications and Networking (BlackSeaCom)*, pp. 107 – 111, August 2013.
- [108] L. Sun and L. Xu, “A Joint Energy-Saving Mechanism for M2M Communications in LTE-based System,” *Wireless Communications and Networking Conference (WCNC)*, pp. 4706 – 4711, April 2013.
- [109] T.-C. Juan and S.-E. Wei, “Data-Centric Clustering for Data Gathering in Machine-to-Machine Wireless Networks,” *IEEE Conference on Communication*, pp. 89 – 94, June 2013.

- [110] D. W. K. Ng and R. Schober, “Energy-Efficient Power Allocation for M2M Communications with Energy Harvesting Transmitter,” *IEEE Global Telecommunication Conference*, pp. 1644– 1649, December 2012.
- [111] P.-C. Wang and J.-F. Tsai, “An efficient global optimization approach for solving mixed-integer nonlinear programming problems,” *International Conference on Computers and Industrial Engineering*, pp. 1– 4, July 2010.
- [112] P.-C. Wang and J.-F. Tsai, “Global optimization of mixed-integer nonlinear programming for engineering design problems,” *International Conference on System Science and Engineering*, pp. 255– 259, June 2011.
- [113] M. Abramson and C. Audet, “Convergence of Mesh Adaptive Direct Search to Second Order Stationary Points,” *Society for Industrial and Applied Mathematics*, vol. 17, pp. 606– 619, May 2006.
- [114] L. M. Rios and N. Sahinidis, “Derivative-free optimization: a review of algorithms and comparison of software implementations,” *Springer*, pp. 1247– 1293, July 2012.
- [115] P. Bonami and M. Kilinc, “Algorithms and Software for Convex Mixed Integer Non-linear Programs,” *Springer*, vol. 154, pp. 1– 39, November 2011.
- [116] C. Audet and G. Savarda, “A mesh adaptive direct search algorithm for multiobjective optimization q,” *European Journal of Operational Research*, pp. 545– 556, November 2010.
- [117] C. Audet and J. E. DENNIS, “Mesh Adaptive Direct Search Algorithms for Constrained Optimization,” *IEEE European Wireless Conference*, vol. 17, pp. 188– 217, May 2006.
- [118] E. Kalvelagen, “Some minlp solution algorithms,” pp. 1– 27, May 2009.

- [119] P. Bonami and A. Watcher, “An algorithmic framework for convex mixed integer nonlinear programs,” *Elsevier*, vol. 5, pp. 186–204, May 2008.
- [120] P. Bonami and J. Lee, “More branch-and-bound experiments in convex nonlinear integer programming,” pp. 1644–1649, December 2011.
- [121] J. Konecny and P. Richtarik, “Simple complexity analysis of direct search,” *arXiv preprint arXiv:1410.0390*, 2014.
- [122] M. A. Abramson, C. Audet, J. W. Chrissis, and J. G. Walston, “Mesh adaptive direct search algorithms for mixed variable optimization,” *Optimization Letters*, vol. 3, no. 1, pp. 35–47, 2009.
- [123] C. Audet, J. E. Dennis Jr, and S. Le Digabel, “Globalization strategies for mesh adaptive direct search,” *Computational Optimization and Applications*, vol. 46, no. 2, pp. 193–215, 2010.
- [124] F. H. Clarke, *Optimization and nonsmooth analysis*, vol. 5. Siam, 1990.
- [125] H. Harb and A. Makhoul, “K-Means Based Clustering Approach for Data Aggregation in Periodic Sensor Networks,” *IEEE Conference on on Wireless and Mobile Computing, Networking and Communications*, pp. 435–441, December 2014.
- [126] G. Kumar and H. Mehra, “An Hybrid Clustering Algorithm for Optimal Clusters in Wireless Sensor Networks,” *IEEE Student Conference on Electrical, Electronics and Computer Science*, pp. 1–6, March 2014.
- [127] C. Sergiou and P. Antoniou, “A Comprehensive Survey of Congestion Control Protocols in Wireless Sensor Networks,” *IEEE Communication Surveys and Tutorials*, vol. 16, pp. 1839–1859, November 2014.

- [128] A. S. Saleh and B. M. Ali, “A survey on energy awareness mechanisms in routing protocols for wireless sensor networks using optimization methods,” *Wiley Transaction on Emerging Telecommunication Technologies*, vol. 25, pp. 1184–1207, December 2014.
- [129] P. Shen and C. Li, “Distributed Information Theoretic Clustering,” *IEEE Transaction on Signal Processing*, vol. 62, pp. 3442–3453, July 2014.
- [130] K. Lin and X. Li, “A K-means clustering with optimized initial center based on Hadoop platform,” *IEEE Conference on Computer Science and Education*, pp. 263–266, August 2014.
- [131] L. Guoli and W. Tintang, “The Improved Research on K-Means Clustering Algorithm in Initial Values,” *IEEE International Conference on Mechatronic Sciences, Electric Engineering and Computer*, vol. 16, pp. 2124–2127, December 2013.
- [132] Y. Zhang, “Tree-Based Resource Allocation for Periodic Cellular M2M Communications,” *IEEE Wireless Communication Letters*, vol. 3, pp. 621–624, December 2014.
- [133] H. Dhillon, H. Huang, H. Viswanathan, and R. Valenzuela, “Fundamentals of Throughput Maximization With Random Arrivals for M2M Communications,” *IEEE Transactions on Communications*, vol. 62, pp. 4094–4109, November 2014.
- [134] H. Dhillon, H. Huang, H. Viswanathan, and R. Valenzuela, “On resource allocation for machine-to-machine (M2M) communications in cellular networks,” *IEEE Globecom Workshops*, pp. 1638–1643, December 2012.
- [135] G. L. Ming Cheng, “Overload Control for Machine-Type-Communications in LTE-Advanced System,” *IEEE Communications Magazine*, vol. 50, pp. 38–45, June 2012.
- [136] A. Laya and J. Alonso-Zarate, “Is the Random Access Channel of LTE and LTE-A Suitable for M2M Communications? A Survey of Alternatives,” *To Appear in IEEE Communications Surveys and Tutorials*, pp. 1–12, 2014.

- [137] A.-H. Mohammed and A. Anpalagan, “Base Station Selection in M2M Communication Using Q-learning Algorithm in LTE-A Networks,” *In Proc. IEEE International Conference on Advanced Information Networking and Applications*, 2015.
- [138] S. Bayat, Y. Li, Z. Han, M. Dohler, and B. Vucetic, “Distributed massive wireless access for cellular machine-to-machine communication,” *IEEE International Conference on Communications*, vol. 2, pp. 2767–2772, June 2014.
- [139] H. Wu and C. Zhu, “SFASA: Accelerated S-ALOHA Using Access History for Event-Driven M2M Communications,” *IEEE/ACM Transactions on Networking*, 2013.
- [140] C. Stefanovic and P. Popovski, “ALOHA Random Access that Operates as a Rateless Code,” *IEEE Transactions on Communications*, vol. 61, pp. 4653–4662, November 2013.
- [141] B. L. Mohammed and M. P. D. Grace, “Application of Q-Learning for RACH Access to Support M2M Traffic over a Cellular Network,” *IEEE Proceedings of European Wireless Conference*, pp. 1–6, May 2014.
- [142] B. L. Mohammed and M. P. D. Grace, “Frame based Back-off for Q-learning RACH access in LTE networks,” *Australasian Telecommunication Networks and Applications Conference (ATNAC)*, pp. 176–181, November 2014.
- [143] A. Galindo-Serrano and L. Giupponi, “Distributed Q-learning for Interference Control in OFDMA-based Femtocell Networks,” *IEEE VTC*, May 2010.
- [144] R. Sutton and A. Barto, *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press, 1998.
- [145] H. Safdar, N. Fisal, R. Ullah, W. Maqbool, F. Asraf, Z. Khalid, and A. Khan, “Resource Allocation for Uplink M2M Communication: A Game Theory Approach,” *IEEE Symposium on Wireless Technology and Applications (ISWTA)*, pp. 48–52, September 2013.

- [146] J. Brown and J. Y. Khan, “Predictive Allocation of Resources in the LTE Uplink Based on Maximum Likelihood Estimation of Event Propagation Characteristics for M2M Applications ,” *IEEE Global Communications Conference (GLOBECOM)*, pp. 4965 – 4970, December 2014.
- [147] E. Ahmed and G. Yasser, “BAT: A Balanced Alternating Technique for M2M Uplink Scheduling over LTE,” *IEEE Vehicular Technology Conference*, pp. 1–6, May 2015.
- [148] A. Biral, M. Centenaro, A. Zanella, L. Vangelista, and M. Zorzi, “The challenges of M2M massive access in wireless cellular networks,” *ELSEVIER Digital Communications and Networks*, pp. 1–19, March 2015.
- [149] L. Zheng, S. Guang, and D. Hui, “Research of elevator group scheduling system based on reinforcement learning algorithm ,” *IEEE International Conference on Measurement, Information and Control (ICMIC)*, vol. 1, pp. 606 – 610, August 2013.
- [150] K. Peng-Yong and P. Dorin, “Reinforcement learning approach to dynamic activation of base station resources in wireless networks,” *IEEE Personal Indoor, Mobile Radio Communications*, pp. 3264 – 3268, September 2013.
- [151] G. Serrano and L. Giupponi, “Distributed Q-learning for aggregated interference control in cognitive radio networks,” *IEEE Transactions on Vehicular Technology*, vol. 59, pp. 1823–1834, May 2010.
- [152] C. Wu and M. D. Felice, “Spectrum management of cognitive radio using multi-agent reinforcement learning,” *International Conference on Autonomous Agents and Multi-agent Systems*, pp. 1705–1712, 2010.
- [153] M. Nazir and M. Bennis, “Learning based mechanisms for interference mitigation in self-organized femtocell networks,” *Asilomar Conference*, pp. 1886 – 1890, November 2010.

- [154] M. Benis and D. Niyato, “A Q -learning Based Approach to Interference Avoidance in Self-Organized Femtocell Networks,” *IEEE Globecom*, pp. 706–710, October 2010.
- [155] X. Chen, J. Wu, and Y. Cai, “Energy-Efficiency Oriented Traffic Offloading in Wireless Networks: A Brief Survey and a Learning Approach for Heterogeneous Cellular Networks ,” *IEEE Journal on Selected Areas in Communications*, pp. 627–640, January 2015.
- [156] Y. Chu, P. Mitchell, and D. Grace, “ALOHA and Q-Learning Based Medium Access Control for Wireless Sensor Networks,” *International Symposium on Wireless Communication Systems*, pp. 511–515, August 2012.
- [157] L. Rucco, A. Bonarini, C. Brandolese, and W. Fornaciari, “A bird’s eye view on reinforcement learning approaches for power management in WSNs ,” *IEEE Wireless and Mobile Networking Conference (WMNC)*, pp. 1–8, April 2013.
- [158] N. J. Patel and R. H. Jhaveri, “Detecting packet dropping nodes using machine learning techniques in Mobile ad-hoc network: A survey ,” *IEEE International Conference on Signal Processing And Communication Engineering Systems (SPACES)*, pp. 468–472, January 2015.
- [159] C. Claus and C. Boutilier, “The Dynamics of Reinforcement Learning in Cooperative Multiagent Systems ,” *American Association for Artificial Intelligence*, 2000.
- [160] W. Qiang and Z. Zhongli, “Reinforcement learning model, algorithms and its application,” *IEEE International Conference on Mechatronic Science, Electric Engineering and Computer*, pp. 1143–1146, August 2011.