

RESERVATION FRAME SLOTTED ALOHA FOR MULTI-CLASS IOT NETWORKS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF
MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

By
Mahzeb Fiaz
January 2019

Reservation Frame Slotted ALOHA for Multi-Class IoT Networks

By Mahzeb Fiaz

January 2019

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.



Nail Akar(Advisor)

Ezhan Karaşan

Mehmet Akif Yazici

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

ABSTRACT

RESERVATION FRAME SLOTTED ALOHA FOR MULTI-CLASS IOT NETWORKS

Mahzeb Fiaz

M.S. in Electrical and Electronics Engineering

Advisor: Nail Akar

January 2019

The Internet of Things (IoT) is a promising technology capable of revolutionizing our work and daily lives. ALOHA based medium access schemes are widely used in IoT applications due to their low complexity despite lower throughput figures. In this study, we aim to improve the performance of Frame Slotted Aloha (FSA) for a single hop IoT network without increasing the overall complexity. Duty cycling is a key concept for managing energy consumption of wireless networks with battery powered nodes having maximum duty cycle constraints. The goal of this study is to improve the performance of frame slotted Aloha by exploiting duty cycle patterns in these networks and using reservations in advance. We discuss the system model for a single class IoT network and study via simulations the performance of Reservation Frame Slotted Aloha (RFSA) as compared to FSA, as well as the performance implications of different system parameters related to traffic patterns. With the insight gained from this preliminary study, we next study a multi-class IoT network with nodes belonging to different classes with different duty cycle constraints. Adopting RFSA for such a network requires different schemes for allocating channel resources for each class. We propose several static and dynamic channel allocation schemes based on our traffic model and study their performance as compared to FSA. Static partitioning has better performance for low traffic loads but dynamic partitioning offers better throughput at higher traffic loads. Selection of an appropriate channel allocation scheme can vary according to the load as well as several system parameters of the network.

Keywords: ALOHA, Internet of Things, Frame Slotted ALOHA, R-ALOHA, Channel Allocation .

ÖZET

REZERVASYON TABANLI ÇERÇEVE-DİLİMLİ ALOHANIN ÇOK-SINIFLI IOT AĞLAR İÇİN KULLANILMASI

Mahzeb Fiaz
Elektrik ve Elektronik Mühendisliği, Yüksek Lisans
Tez Danışmanı: Nail Akar
Ocak 2019

Nesnelerin İnterneti (IoT), işimiz ve günlük hayatımızda önemli değişikliklere yol açma potansiyeli olan bir teknolojidir. ALOHA tabanlı ortam erişim yöntemleri, göreceli düşük verime sahip olmalarına rağmen, düşük karmaşıklıklarından dolayı, IoT uygulamalarında yaygın olarak kullanılmaktadır. Bu çalışmada, tek sekmeli bir IoT ağı için Çerçeve-Dilimli ALOHA'nın (FSA) başarımını arttırmak hedeflenmektedir. Görev zaman dilimleme, kablosuz ağların enerji tüketimini yönetmek için önemli bir kavram olarak ortaya çıkmaktadır ve lisanssız spektrumda çalışan ağları kullanan düğümlerde bir maksimum görev döngüsü kısıtlaması mevcuttur. Bu çalışmanın amacı, bu ağlarda görev döngüsü paternlerini kullanmak suretiyle önceden rezervasyon yapılarak FSA'nı başarımını arttırmaktır. Öncelikle, tek-sınıflı bir IoT ağı için sistem modeli önerildi ve simülasyon yöntemi kullanılarak, önerilen Rezervasyon FSA (RFSA) yönteminin, FSAYe göreceli olarak, trafik modelleri ile ilgili farklı sistem parametrelerindeki değişikliklere göre başarımı irdelendi. Bu ön çalışmadan elde edilen bilgiler ışığında, farklı görev döngüsü kısıtlamaları olan çok sınıflı bir IoT ağı için RFSA yöntemleri önerildi. Çok sınıflı bir IoT ağında RFSA'nın kullanılması için, her bir sınıf için kanal kaynak tahsis yöntemleri gerekir. Bu tezde, kullanılan trafik modeline göre çeşitli statik ve dinamik kanal yöntemleri önerilmektedir FSA ile karşılaştırmak suretiyle, bu yöntemlerin başarımları irdelenmektedir. Statik bölümlleme, düşük trafik yükleri için daha iyi başarıma sahiptir, ancak dinamik bölümlleme, daha yüksek trafik yüklerinde daha iyi bir performans sunmaktadır. Uygun bir kanal tahsisi yönteminin seçiminin, yüke ve ağın çeşitli sistem parametrelerine göre değişmesi gerektiği, bu tezde gösterilmiştir.

Anahtar sözcükler: ALOHA, Nesnelerin Interneti, Çerçeve-Dilimli ALOHA, R-ALOHA, Kanal Tahsisi.



Acknowledgement

First and foremost, I would like to express my sincere gratitude to my supervisor Prof. Nail Akar for his constant guidance, support and encouragement. This work would not have been possible without his guidance.

I would like to thank Prof. Ezhan Karasan and Dr. Mehmet Akif Yazıcı for agreeing to be on my thesis committee and for their valuable feedback.

I would like to acknowledge the TÜBİTAK support of my research as a scholar in ARDEB-1001 project number 115E360.

I would like to thank my parents for providing me with the best of opportunities and for their unconditional support. I would like to dedicate this thesis to my late grandmother without whom I would not be the person I am today.

Last but not the least, I would like to thank my siblings and friends for all the love and support they offered during the course of my degree.

Contents

1	Introduction	1
1.1	Overview	1
1.2	Literature Review	5
1.3	Thesis Contribution	9
1.4	Thesis Outline	10
2	Related Work	11
2.1	Pure ALOHA	11
2.2	Slotted ALOHA	13
2.3	Frame Slotted ALOHA	14
2.4	Reservation Frame Slotted ALOHA	15
3	Reservation Frame Slotted ALOHA for Single-Class Network	17
3.1	System Model	18
3.2	Performance Metrics	20

3.3	Simulation Results	21
3.3.1	Throughput	23
3.3.2	Delay	24
3.3.3	Energy Consumption	25
3.3.4	Packet Drop Probability	25
3.3.5	Delay and Throughput Comparison for varying A , B and M	26
4	Reservation Frame Slotted ALOHA for Multi-Class Network	33
4.1	System Model	33
4.2	Performance Metrics:	35
4.3	Algorithms:	36
4.3.1	Reservation Frame Slotted ALOHA-SP	37
4.3.2	Reservation Frame Slotted ALOHA-SPS	37
4.3.3	Reservation Frame Slotted ALOHA-DP	38
4.4	Simulation Setup:	40
4.5	Simulation Results:	40
4.5.1	Reservation Frame Slotted ALOHA-SP	40
4.5.2	Reservation Frame Slotted ALOHA-SPS	43
4.5.3	Reservation Frame Slotted ALOHA-DP	49
4.5.4	Static and Dynamic Partitioning Throughput Comparison	52

4.5.5 Static and Dynamic Partitioning with Unequal Channel Al-
location 54

4.5.6 Performance of Static and Dynamic Partitioning for vary-
ing traffic patterns 56

5 Conclusions **61**



List of Figures

1.1	Internet of Things paradigm as a result of the convergence of different visions [1]	2
1.2	Single hop Internet of Things network	4
1.3	Internet of Things applications [2]	6
2.1	Packet Transmission Example pure ALOHA	12
2.2	Packet Transmission Example slotted Aloha	13
2.3	Packet Transmission Example FSA	14
2.4	Packet Transmission Example R-ALOHA	16
3.1	Traffic Pattern for nodes	18
3.2	Throughput T RFSA vs FSA as function of number of nodes N	23
3.3	Average delay D per successfully transmitted packet RFSA vs FSA as function of number of nodes N	24
3.4	Number of retransmissions R per packet RFSA vs FSA plotted as function of number of nodes N	25

3.5	Packet Drop Probability P_D RFSA vs FSA as function of number of nodes N	26
3.6	Throughput T for RFSA with respect to varying value of mean burst length B	27
3.7	RFSA average delay D with respect to varying value of mean burst length B	28
3.8	Throughput T for RFSA with respect to varying value of mean idle time A	29
3.9	RFSA average delay D with respect to varying value of mean idle time A	30
3.10	Throughput T for RFSA with respect to varying value of number of channels M	31
3.11	RFSA average delay D with respect to varying value of number of channels M	32
4.1	System Model	34
4.2	Throughput T as a function of load ρ for FSA and RFSA-SP. . .	41
4.3	Blocking probability P_b as a function of load ρ for FSA and RFSA-SP. . .	42
4.4	Number of retransmissions R per packet as a function of load ρ for FSA and RFSA-SP.	43
4.5	Throughput T as a function of load ρ for FSA and RFSA-SPS ₄ . .	44
4.6	Blocking probability P_b as a function of load ρ for FSA and RFSA-SPS ₄	45

4.7	Number of retransmissions R per packet as a function of load ρ for FSA and RFSA-SPS ₄	46
4.8	Throughput T as a function of load ρ for FSA and RFSA-SPS ₈	47
4.9	Blocking probability P_b as a function of load ρ for FSA and RFSA-SPS ₈	48
4.10	Number of retransmissions R per packet as a function of load ρ for FSA and RFSA-SPS ₈	49
4.11	Throughput T as a function of load ρ for FSA and RFSA-DP.	50
4.12	Blocking probability P_b as a function of load ρ for FSA and RFSA-DP.	51
4.13	Number of retransmissions R per packet as a function of load ρ for FSA and RFSA-DP.	52
4.14	Throughput T as a function of load ρ for RFSA-SP, RFSA-SPS and RFSA-DP.	53
4.15	Throughput T as a function of load ρ for RFSA-SP and RFSA-SPU.	55
4.16	Throughput T as a function of load ρ for RFSA-DP and RFSA-DPU.	56
4.17	RFSA-SP and RFSA-DP throughput T as a function of α for $\rho_3 = 1/2$	58
4.18	RFSA-SP and RFSA-DP throughput T as a function of α for $\rho_3 = 1/3$	59
4.19	RFSA-SP and RFSA-DP throughput T as a function of α for $\rho_3 = 1/4$	60

List of Tables



Chapter 1

Introduction

1.1 Overview

With the advent of the Internet of Things (IoT) and Machine-to-Machine (M2M) communications, the number of devices connected to the Internet is increasing rapidly. Sensors, actuators and RFID technologies work together to seamlessly integrate to the Internet. Advancements in Micro-Electronic-Mechanical Systems (MEMS) technology, digital electronics and wireless communications have resulted in the development of small devices capable of sensing, computing and communicating over small distances. A network of these nodes is called a Wireless Sensor Network (WSN) with a wide range of applications in industrial monitoring, environment monitoring, infrastructure monitoring, retail; see [3] and the reference therein. In [1], the authors define three paradigms for the IoT: internet-oriented (middleware), things-oriented (sensors) and semantic-oriented (knowledge). The true potential of the Internet of Things can only be unleashed in application domains where these paradigms intersect.

Access to shared channel resources is an important concern for wireless networks. Throughput of the underlying medium access schemes can significantly affect the overall performance of the network. Therefore, careful design of medium

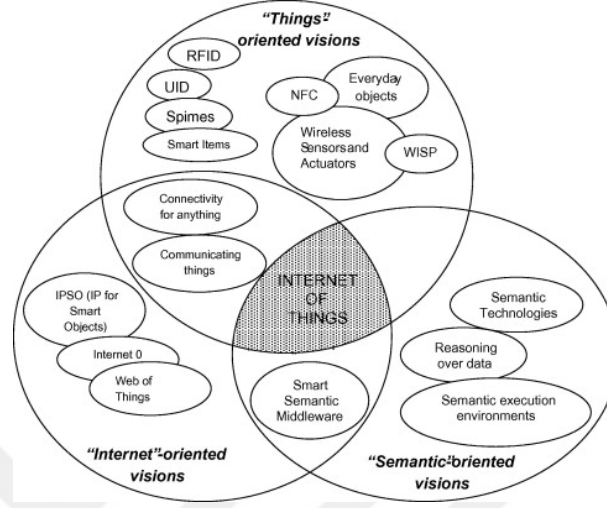


Figure 1.1: Internet of Things paradigm as a result of the convergence of different visions [1]

access schemes is vital to optimal operation of the network. Medium access schemes proposed for wired networks cannot be used for wireless networks without modification. A number of medium access schemes have been proposed for wireless networks in the literature. MAC protocols for wireless networks can be categorized into scheduled and event-driven protocols [4]. Schedule-based MAC protocols like TDMA, FDMA, CDMA avoid collisions by reserving resources for each node. Network resources, however, are used wastefully as every node may not use the resource reserved for it at all times. Another drawback of these protocols is that they are not adaptable and scalable. Low Energy Adaptive Clustering Hierarchy (LEACH) [5] and Bluetooth [6] are examples of scheduled protocols. In LEACH, nodes are organized into clusters with TDMA within each cluster managed by a cluster head. The role of the cluster head is rotated between nodes in the cluster to avoid uneven energy depletion. Bluetooth standard was introduced for Personal Area Networks (PAN) and organizes nodes into piconets. Piconets are clusters consisting of a master node and upto seven slave nodes. For communications inside the cluster, TDMA is used and for inter-cluster communications, frequency-hopping CDMA is used.

Event-driven protocols do not allocate channel resources for nodes in advance; instead resources are allocated to nodes when they have data to send. While

these protocols are more scalable and adaptable to changes in the network, for example the node density and traffic load, they are also prone to collisions. Other drawbacks of these protocols include idle listening and over-hearing [7]. ALOHA and its predecessors and CSMA/CA are the most prominent protocols in this category.

Introduced in 1970, ALOHA [8] was the first of its kind medium access system that provided wireless communication between different campuses of the University of Hawaii. Aloha considers a star topology with a single gateway communicating with multiple nodes. Each node performs random access without any prior knowledge about the channel to transmit its data. If only one node transmits during this time, transmission is successful, otherwise, a collision occurs. While Aloha offers an attractive medium access mechanism due to its low complexity its throughput is also low due to collisions. A number of variations of pure Aloha were later introduced to improve its throughput performance. Slotted Aloha (S-Aloha) [9] is one such scheme which divides time into slots and transmissions are synchronized according to these time slots. Each node can only transmit at the start of a time slot. This synchronization results in a reduced number of collisions and hence improves the performance of pure Aloha. Frame Slotted Aloha (FSA) [10] is another predecessor of pure Aloha which divides time into frames. Each frame consists of multiple slots. Each node randomly selects a slot for transmitting its data in a frame. Reservation Aloha (R-Aloha) [11] was proposed to improve the performance of FSA for satellite communications where long messages are fragmented into smaller packets each of which can be transmitted in one slot. It was proposed that in these scenarios if a node successfully transmits its first packet in a slot, this slot should then be reserved for this node at the successive frames until the end of the message. This protocol improves the performance of FSA without increasing the overall complexity. While it was initially proposed for satellite communications, it also found applications in vehicular networks, optical networks, and other domains [12–14].

For IoT networks and M2M communications, energy conservation is one of the main concerns and hence MAC protocols need to be simple and efficient. While a number of protocols are proposed for these networks in the literature, in

practical applications, Aloha-based protocols are widely used. In a single-hop IoT network, nodes communicate with a single gateway with no direct communication taking place between the nodes. In some of these networks, nodes transmit longer messages fragmented into packets. For such networks, R-Aloha can be used as it improves the throughput without increasing the complexity.

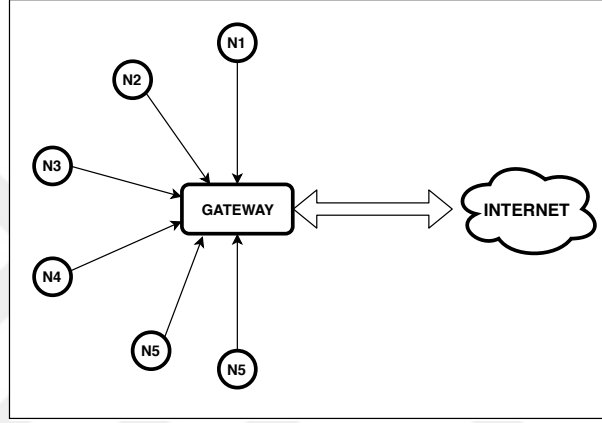


Figure 1.2: Single hop Internet of Things network

R-Aloha has been previously studied for M2M communications and is often referred to as Reservation Frame Slotted Aloha (RFSA). Most of these studies focus on its use as an anti-collision algorithm for tag detection in RFID and merely delta traffic conditions have been considered. IoT networks, however, are very diverse with a variety of applications with unique system and traffic models and there is a gap in the literature in terms of studying RFSA for different IoT networks. Keeping in mind this gap, we aim to study the performance of RFSA for single hop IoT networks with duty cycling. In addition, we adapt this protocol for a multi-class heterogeneous IoT network.

We study the throughput, delay and energy performance of RFSA for a duty-cycled network in which nodes have alternating idle and busy periods. During busy periods, the nodes transmit a burst of data with packets while during idle periods the transmitters are turned off. Through simulations, we compare the performance of RFSA with FSA and show that it outperforms FSA in terms of our performance metrics. In addition, we study the effect of system parameters on the throughput and delay performance of RFSA. Building on the insights gained

through these results, we then introduce a heterogeneous IoT network with nodes belonging to different classes. Nodes in class i sample data packets with a period P_i during the busy period and transmit them. As each class has a unique period, the use of RFSA in such a network requires channel allocation schemes for distributing channel resources between these classes. We propose three different channel allocation schemes in the context of two reservation-based and one sporadic traffic class and study their performance. More than two reservation classes of traffic is left for future research.

1.2 Literature Review

Internet of Things (IoT) is an extension to the traditional Internet which promises to let a massive number of Internet-enabled devices to exchange sensed data over the Internet. While IoT is a promising technology with substantial potential to revolutionize both our daily lives and the way we work, it poses several design challenges. The end devices in an IoT network are small battery powered nodes with very limited energy and computational resources.

Internet of Things is still considered an emerging technology but Internet-enabled devices were available in the 1990s. John Romkey [15] introduced the first known IoT device, an Internet controlled toaster in 1989. This toaster could be turned on/off using the Internet. In 1999, the term Internet of Things (IoT) was introduced by Kevin Ashton [16], executive director of the Auto-ID center, in the context of supply chain management. However, in the past decade, the definition has widened with IoT finding applications in all venues of life including health care, utilities, transport, etc. [17]. It would be appropriate to say that IoT has stepped out of its budding stage and is on the verge of enabling the fully integrated Future Internet [2]. In 2008, Cisco reported that there were more devices connected to the Internet than people and it is estimated that by 2020, there will be over 50 billion IoT devices [18]. IoT is finding applications in every aspect of our lives and soon Internet will be deeply woven into the fabrics of our daily lives blurring the boundaries between cyber and physical worlds.

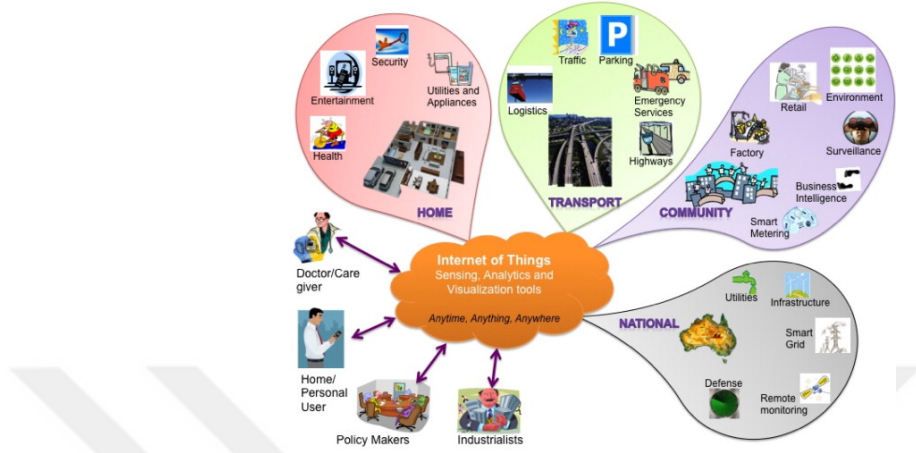


Figure 1.3: Internet of Things applications [2]

IoT networks operating in the license free spectrum are subject to a variety of regulations by different regulatory authorities. One of these regulations is the maximum duty cycle limit per device. This limit has been defined to prevent a transmitter from occupying a channel for extensive period of time. The maximum duty cycle limit is in the range of 2.8 percent for devices complying with Listen Before Talk and Adaptive Frequency Agility [19]. In addition to the maximum duty cycle constraint, duty cycling is also an important concept for conserving energy of the network. A number of duty cycling algorithms [20–25] have been proposed in the literature to manage the energy consumption of nodes.

Internet of Things networks generally consist of low-power devices with low computational complexity. These devices have limited energy and computational resources, making energy efficiency the main design challenge. This tradeoff between efficiency and energy consumption has led to substantial research for designing efficient medium access schemes for these networks. A number of MAC protocols have been proposed for IoT networks in the past. MAC protocols for IoT can be categorized into contention-based protocols, contention-free protocols and hybrid protocols [26].

Aloha [8] was the first random access scheme that proposed transmission of messages as they arrive without sensing the channel. Through the use of positive acknowledgments and checksums, this protocol allows the nodes to transmit data

in an unsynchronized manner. Due to lack of central control, there is a probability of collisions in this protocol due to which it cannot utilize the channel resources to the fullest extent. The maximum achievable throughput of Aloha is $1/2e \approx 0.18$.

To improve the performance of pure Aloha, slotted Aloha [9] was proposed. In slotted Aloha, the time axis is divided into identical slots. Size of these slots is chosen to be equal to the transmission time of a single packet. Each node can access the channel at the start of a slot with a certain probability. Since there are no transmissions during a slot, the number of collisions is significantly reduced thus resulting in doubling of the throughput of pure Aloha. Frame slotted Aloha [10] is another variant of slotted Aloha which further divides into frames which are composed of a number of time slots. Each user can transmit data in a randomly chosen time slot and the decision is to be made on a per-frame basis.

Slotted Aloha and frame slotted Aloha-based protocols have found a wide range of applications in networked systems due to their low complexity and effectiveness in tackling collisions. These applications range from satellite communications [27] and wireless local area networks [28] to more recent applications such as M2M communications [29]. Frame slotted Aloha has gained a lot of attention for tag identification in RFID systems [30], [31] and is also a part of the EPCGlobal Class-1 Generation-2 (C1G2) RFID standard [32].

The work in [11] proposes R-Aloha for satellite communications with the aim of maximizing channel utilization with a low complexity medium access scheme. In satellite communications, nodes need to transmit long messages comprising multiple packets over a shared broadcast channel. The reference [33] provided a performance analysis of R-Aloha for packet broadcast networks and provides an analytic framework for characterizing the message delay and channel utilization of the protocol. A number of expressions for channel throughput and channel capacity are derived and it is shown that the channel capacity of R-Aloha ranges from that of slotted Aloha to one, depending on the input traffic and user model.

While R-Aloha was initially proposed for satellite communications making use of the unique traffic patterns of a satellite network, it also found a number

of applications in other types of communication networks. In vehicular adhoc networks, R-Aloha based medium access schemes offer an attractive alternative to contention-based schemes by offering a solution to mobility issues without compromising on the QoS metrics. In [13], Reliable R-Aloha (RR-Aloha) was proposed for ad-hoc networks with hidden terminal problems. Later RR-Aloha+ [14], an extended version of the protocol was proposed which improves its stability and reliability without increasing its complexity. Other variations of RR-Aloha have been proposed and analyzed in the literature [34], [35] for improving its resilience to mobility issues facing VANETs. Reservation-based Aloha protocols offer an attractive solution for VANETs.

In [12], the authors study the use of S-Aloha based protocols, including R-Aloha for optical fiber local area networks. The study concludes that R-Aloha protocols are best suited for bulk data transfers in optical fiber networks and offer better throughput and delay performance than slotted Aloha. While initially the term R-Aloha was used, later many studies referred to the protocol or its variants with different names. In [36], the performance of Packet Reservation Multiple Access (PRMA) is studied with PRMA being a close relative of R-Aloha.

R-Aloha has also gained attention of researchers in IoT domain for M2M communications. Most of the literature in this domain refers to R-Aloha as Reservation Frame Slotted Aloha (RFSA). In [37], the authors study the performance of RFSA for data collection in M2M networks. In such networks, the nodes keep their radios off until prompted by a coordinator to transmit their sensed data. The study concludes that RFSA outperforms FSA in terms of its throughput, delay and energy performance for both the coordinator and the end nodes. In [38], the authors provide an analysis of RFSA for delta traffic conditions in which all nodes compete to access the channel at the same time setting the network into saturation. They derive an optimal frame length to maximize throughput and delay performance depending on the number of active end nodes.

Tag collision is an important issue in IoT networks which results in low recognition during population process of an IoT network. Aloha based protocols have also found application as anti-collision algorithms for tag identification. In [39],

the authors proposed a dynamic frame slotted Aloha (DFSA) which dynamically adjusts the number of frames in a slot according to number of tags. It overcomes the shortcomings of other Aloha-based protocols when the number of tags is large. In [40], an improvement to the DFSA algorithm has been proposed which outperforms DFSA in certain situations.

1.3 Thesis Contribution

In the literature, RFSA has been studied for IoT networks but these studies are limited to its use in tag reading and other delta traffic conditions. In delta traffic conditions, all nodes go from idle to busy mode and attempt to access the channel at the same time. While these studies conclude that RFSA outperforms FSA for such networks, there has been no work done in the past to evaluate the performance of RFSA for other IoT networks with different classes of traffic.

In this thesis, we performed a simulation study to evaluate the throughput, delay and energy performance of RFSA for IoT networks with nodes having alternating *ON* and *OFF* periods. During an *OFF* period the nodes remain idle and keep their radios off. During an *ON* period, on the other hand, the nodes turn on their radios and transmit a burst of data. Such a traffic model can be a realistic representation of IoT networks operating in the unlicensed spectrum where the nodes have a duty cycling constraint. In a duty cycled network nodes have an upper bound on the amount of time they can transmit. We compare the performance of RFSA with FSA in such a network.

In IoT networks, as we know there are a number of different devices working together and hence many IoT networks will not necessarily be homogeneous. In a heterogeneous network, not all nodes belong to the same class. Hence in a realistic representation of an IoT network, there are multiple classes. For using RFSA in a multi-class IoT environment, one is faced with the challenge of distributing the network resources among these classes. We propose three different schemes for dividing channel resources and compare their performance. So, the main novel

contribution of this thesis is adaptation of RFSA for a multi-class IoT network.

1.4 Thesis Outline

This thesis is organized as follows: In Chapter 2, we give some important preliminaries required for thorough understanding of the thesis. We discuss pure Aloha, slotted Aloha and reservation frame slotted Aloha in detail. In Chapter 3, we give a detailed overview of our system model and give the simulation results of RFSA as compared to FSA for our traffic model. In Chapter 4, we first describe our system model with two reservation-based classes and one sporadic class. Each node within a reservation-based class transmits with a period P_i during busy periods and the load on the system is a-priori known. We propose three different channel allocation schemes for this system model and study their performance as compared to FSA. We also study their performance for a wide range of system parameters and show that the choice of a channel allocation mechanism depends on the parameters of the traffic model. Finally, in Chapter 5, we present our conclusions and discuss future research directions.

Chapter 2

Related Work

This chapter provides a detailed background on the basic concepts required for understanding this thesis. First, we provide a detailed overview of pure Aloha with mathematical analysis and throughput. Next, we discuss slotted Aloha and frame slotted Aloha in detail. Frame slotted Aloha serves as our performance benchmark throughout this simulation study. And last we discuss reservation frame slotted Aloha as it forms the basis for the simulation study.

2.1 Pure ALOHA

Abramson proposed Aloha [8] as the first packet communication system designed to provide wireless access to computer systems. In 1971, the Alohanet became operational which connected different campuses of the University of Hawaii. The medium access scheme used by this network was referred to as the Aloha protocol which gained a lot of attention due to its simple nature. Aloha laid down the foundation for random access schemes which are widely studied in the literature. The simple and distributed nature of Aloha has helped in keeping it relevant to date and its variations are widely adopted in modern communication systems.

Aloha considers a simple one-hop network in which each node communicates

with a common gateway. Each node does not have any prior information about channel usage and transmits a packet right away as it arrives. If multiple nodes try to transmit at the same time, a collision occurs, otherwise transmission is successful. For each correctly received packet, the gateway sends an acknowledgment. If a node does not receive acknowledgment, it re-transmits the packet.

In Aloha, T_s is the transmission time of a single packet. If we consider the time of transmission for a packet to be t , a collision occurs if any other node transmits within the $t - T_s$ to $t + T_s$ time interval. Since each node can transmit at any given time so the probability of collisions is high in pure Aloha. Fig. 2.1 shows an example of such a network with 5 users. For the 5th user since the third user started transmission within $t - T_s$, so there is a collision. Similarly, for the third user, the second packet also collides as the 5th user starts transmitting within the $t + T_s$ interval.

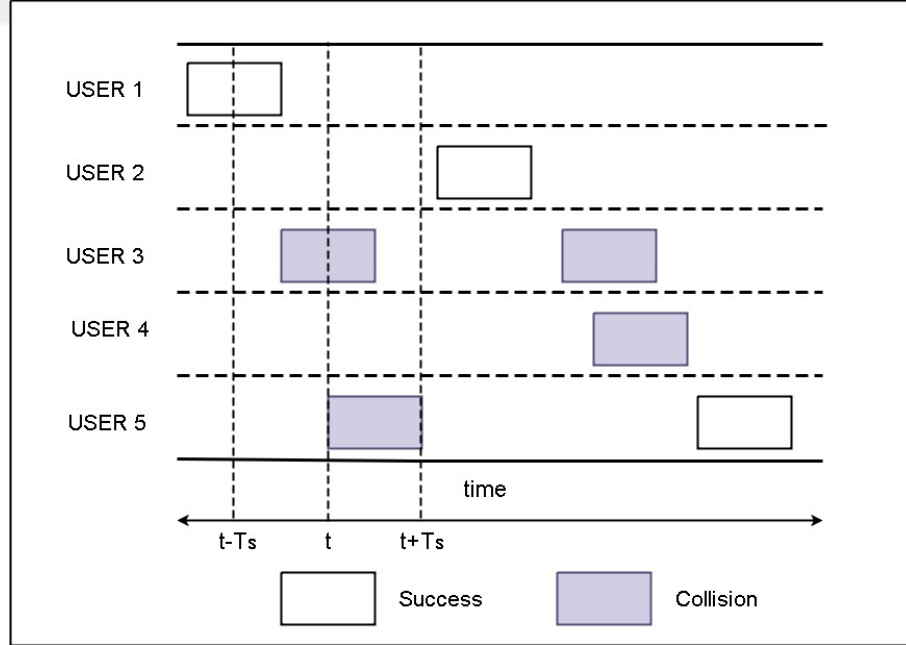


Figure 2.1: Packet Transmission Example pure ALOHA

Throughput T for this system can be defined as the number of successfully received packets per unit time, where a unit time can be taken as T_s . We can define load G on the system as the number of packets transmitted per unit time. If we consider arrivals to be Poisson, then the throughput of the system is $T =$

Ge^{-2G} . The maximum throughput achieved by Aloha is at $G = 0.5$ and is $1/2e \approx 0.18$ [8]. Since the throughput of pure Aloha is very low, several variations were proposed to improve its throughput.

2.2 Slotted ALOHA

Slotted Aloha (S-Aloha) is one variation of pure Aloha which aims to improve the maximum achievable throughput through time synchronization. In S-Aloha time is divided into slots equal to the transmission time of a single packet, i.e., T_s . All nodes are synchronized by the gateway and can only transmit a packet at the start of a time slot. In such a network, the probability of collisions goes down as nodes cannot start a transmission at any random time instant hence resulting in no partial collisions. Collisions only occur if another node starts transmission at the same time slot as this node. In Fig. 2.2, an example network for S-Aloha is given. A collision occurs for the 3rd and 5th user as they both attempt to transmit in the same time slot. For all other users, transmission is successful as no other node transmits in the same time slot.

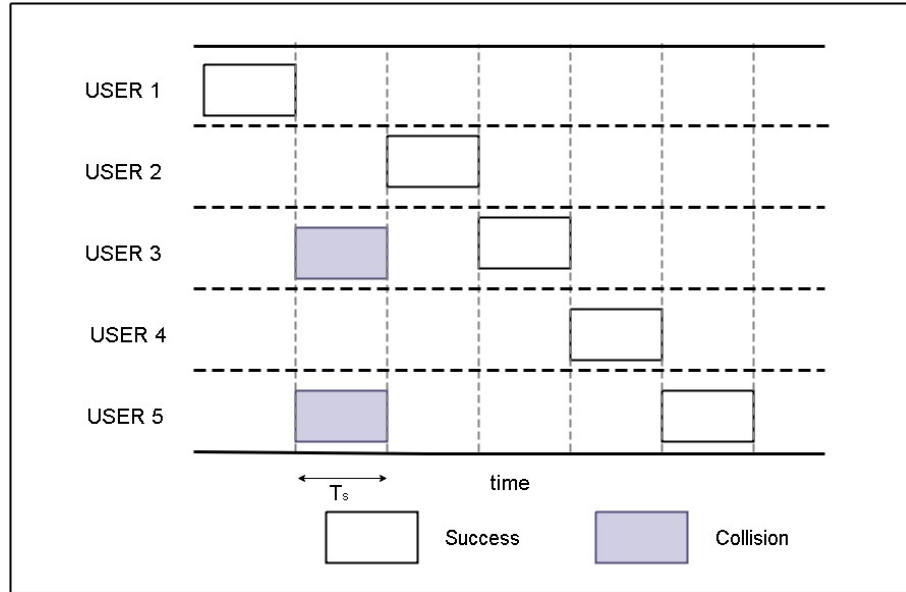


Figure 2.2: Packet Transmission Example slotted Aloha

S-Aloha increases the complexity of the system as it requires synchronization but also improves the throughput of pure Aloha. For a system throughput T is defined as the number of successfully received packets per time slot and load G is defined as the number of packets transmitted per time slot. For slotted Aloha, $T = Ge^{-G}$ which is maximized at $G = 1$ as $1/e \approx 0.368$ [9]. While the throughput for slotted Aloha is significantly higher than pure Aloha, more than 60 percent of the network time is still being wasted.

2.3 Frame Slotted ALOHA

Frame slotted Aloha (FSA) is another variation of the pure Aloha protocol. In FSA, also referred to as Framed Aloha, time is divided into frames with each frame further divided into multiple slots. FSA is similar to S-Aloha in terms of synchronization. Nodes can only start transmission at the start of a time frame. Each slot in a time frame has the size T_s . Each node can transmit in one slot per frame. Whenever a packet arrives, the node randomly selects a slot to transmit this packet in the current frame. A collision occurs only if more than one node tries to transmit a packet in the same slot in the same frame.

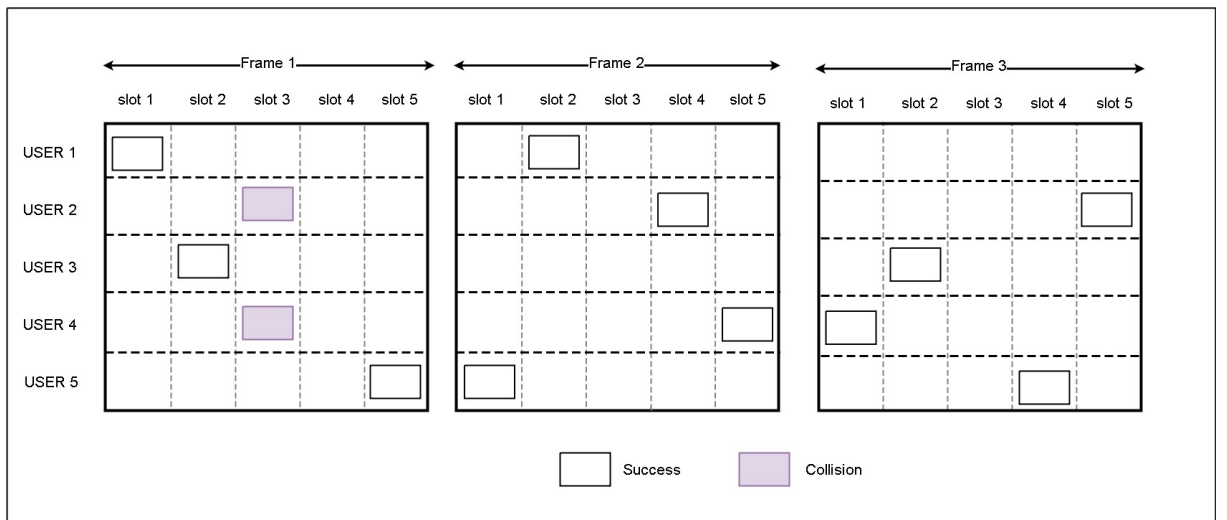


Figure 2.3: Packet Transmission Example FSA

Fig. 2.3 gives an example of a sample network with FSA. Each node randomly

selects a slot in a frame to transmit its packet. In Frame 1, there is one collision in slot 3 as the 2nd and 4th user transmit in it. Transmissions are successful in all other slots in all frames as only one user transmits. FSA lowers the probability of collisions by providing multiple slots within each frame to the users.

Frame slotted Aloha initially found applications in satellite communication but was later adopted for a number of communication networks including RFID tag recognition due to its low complexity and high throughput. An interesting problem in the FSA protocol is the selection of the number of slots available in a frame depending on the number of active users. A number of studies have been conducted in this regard and dynamic versions of the protocol have been proposed which optimize the number of slots available in a frame to maximize throughput.

2.4 Reservation Frame Slotted ALOHA

Reservation Aloha was proposed in 1973 with the aim to maximize channel utilization in satellite networks. In Satellite Communications generally, long messages are transmitted consisting of multiple consecutive packets. The idea of RFSA is that for such communications, a node should contend for a slot in a frame only for the first packet in a long message. Once it successfully captures a slot for transmission, the slot should be reserved for this node for the entire duration of this message. So RFSA behaves as a Random Access (RA) protocol for the first packet, but for the rest of the packets in the message it behaves as TDMA.

In R-Aloha each node has information about the usage status of each slot in the previous frame. From a node's perspective, each slot can either be used which means it was used successfully for transmission or unused which means it was either not used by any node or a collision took place. If a slot was used by node X in the previous frame it is marked as forbidden for all other nodes and node X will continue its transmission in this slot. The unused slots in the last frame are available for contention by all the nodes. There can be two variations

for this protocol, P1 with no end of usage flag and P2 with the end of usage flag. P2 offers a better throughput as no slots are wasted but P1 is simpler in terms of implementation.

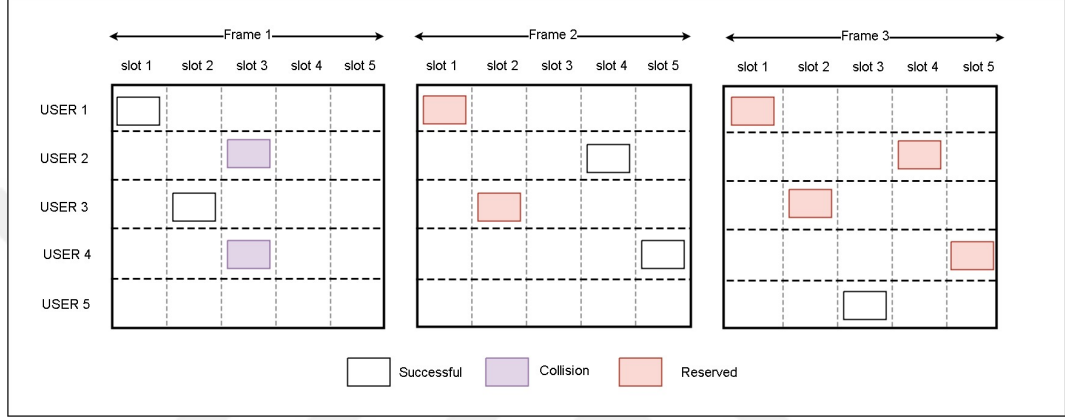


Figure 2.4: Packet Transmission Example R-ALOHA

Fig. 2.4 provides a packet transmission example of R-Aloha. The users which manage to transmit a packet successfully in a slot in the given frame, reserve it for transmission in the subsequent frames. A reserved time slot is forbidden for all other users and cannot be used for transmission. As there is a collision for 2nd and 4th users in the first frame, no reservations are made by them. They transmit again in the second frame and as they are successful, they reserve the time slots for the third frame.

In [33] author provided a detailed analysis of R-Aloha protocol for packet broadcast networks in 1980. In this analysis, author concludes that R-Aloha adapts itself to the nature of input traffic. The throughput of R-Aloha can range from that of slotted Aloha to 1, depending on the input traffic. While [33] provides an analysis of the protocol for two different user models, it also mentions some interesting problems that might arise in practical implementations of the protocol. In conclusion, author discusses non-homogeneous networks in which nodes can belong to different classes and how in such scenarios analysis may not be possible.

Chapter 3

Reservation Frame Slotted ALOHA for Single-Class Networks

In certain IoT applications, the traffic pattern is bursty with nodes having alternating *OFF* and *ON* times. They transmit a burst of data during the *ON* time and remain idle during *OFF* time. For such systems in which end devices need to send a burst of packets, we can enhance the performance of FSA by reserving a time slot for a node after it successfully secures a channel. The channel remains reserved for this node until the entire burst is transmitted. This scheme can be easily implemented through a forbidden list of reserved channels broadcast by the gateway. With this information, nodes can randomly select a channel from the ones not included in the forbidden list.

In this chapter, we present a system model for such a network and compare the performance of RFSA and FSA through simulations. Next, we modify different system parameters and study their effect on our performance metrics.

3.1 System Model

We consider a single hop network with a star topology. In the proposed network a single gateway communicates with multiple end nodes. Gateway is considered to have unlimited power and computational resources, however, the end nodes are battery powered with limited computational capability. No communication takes place between the end nodes. Time is divided into slots and in each time slot, multiple channels are available.

The nodes in this system model have alternating idle and busy periods. At the end of an idle period, a burst of data with multiple packets is generated. Packet sizes are such that a single packet is transmitted during a single time slot. For each node, a cycle consists of one *ON* period followed by an *OFF* period. During the *ON* period, a node tries to transmit the burst of data generated at its start.

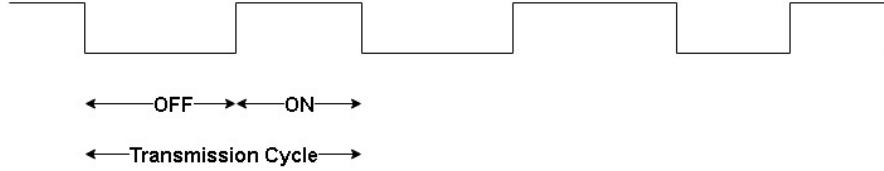


Figure 3.1: Traffic Pattern for nodes

The system parameters are as follows:

- Mean burst length B ,
- Mean idle time A ,
- Number of channels per time slot M ,
- Number of nodes N .

A homogeneous network comprising N nodes is considered such that all nodes have the same traffic characteristics. Burst length and idle times are geometrically

distributed with mean B and A , respectively. At the end of each idle time, a node wakes up and generates a burst of data. It now contends to secure a channel for transmitting this burst. If we define $E[ON]$ as the average number of time slots required for the transmission of a burst, we can write the packet generation rate R as:

$$R = \frac{B}{A + E[ON]} \quad (3.1)$$

While we have fixed the mean burst length and mean idle period, $E[ON]$ is not fixed and hence the packet generation rate for each node is not fixed. A node can reserve only one channel within a time slot. If a node successfully transmits a packet on a channel, that channel is reserved for this node in the subsequent slots and is marked as forbidden for all the other nodes. The channel selection algorithm is give below:

Algorithm 1: Channel Selection

```

if First Packet of the burst then
  if all channels not busy then
    Repeat:
      Randomly select channel
      if Channel not FORBIDDEN then
        Increment BUSY for selected channel
        Set CURR_CHANNEL to selected channel
      else
        goto Repeat:
      end if
    else
      Start BACKOFF Timer
    end if
  end if

```

As we consider the end nodes as resource constrained, we keep the complexity

for channel selection algorithm low. Each channel is selected with an equal probability. If a node wants to transmit but all the channels are reserved, it starts a *BACKOFF* timer which is one half the mean idle time A . Once the *BACKOFF* timer ends, it once again contends to reserve a channel for transmission. If the delay experienced by the first packet in the burst exceeds the mean idle time A , the burst is dropped and the node starts its *OFF* timer. We know that for RFSA a node can only go into backoff for the first packet in the burst, however, for FSA node can go into backoff for any packet in the burst. So for RFSA an entire burst is dropped, but for FSA a burst can be dropped in the middle. At the end of each time slot, the gateway broadcasts a Forbidden List of all the reserved channels.

3.2 Performance Metrics

In this section, we provide details of the performance metrics we will be using in this chapter to evaluate the performance of different medium access schemes.

The first and most important performance metric we will be using is the throughput T . We define throughput T as the number of successfully transmitted packets per channel where each time slot consists of M channels.

Our second performance metric is delay D and we define delay D as the average delay per successfully transmitted packet.

We will also analyze the packet drop probability P_D for each protocol, where P_D is the number of packets dropped per generated packet.

Our last performance metric is the average energy consumed by a protocol. We assume that for each node E_{tr} is the energy consumed for transmitting each packet and R is the number of retransmissions per packet generated in the network. We take the average Energy consumption of the network to be $E = E_{tr} \times R$. Since the energy consumption is directly related to the number of retransmissions per packet so we take it to be our metric of interest.

3.3 Simulation Results

We have implemented our simulator in C++. Our simulator consists of three classes: *Gateway*, *SGateway*, and *Node*. *SGateway* class runs the simulation for our system model and given input parameters for FSA. *Gateway* class does the same for RFSA. We create objects of both *Gateway* and *SGateway* class. The constructor for each class takes the number of nodes N , the number of channels M , mean burst length B and mean idle time A as input. In each constructor, N objects of the *Node* class are created and initialized with the given parameters. The constructor then calls the run simulation function which takes the *Node* objects as input. Pseudo code for this function is given below:

Algorithm 2: RunSimulation

```
while SimTime  $\leq T_{max}$  do
  Set BUSY to zero for all channels
  for all Nodes do
    if BACKOFF timer=0 then
      if OFF timer =0 then
        Set new ON timer
      else
        if OFF timer  $\neq$  0 then
          Decrement OFF timer
        else
          if First Packet of the burst then
            if all channels not busy then
              Repeat:
                Randomly select channel
                if Channel not FORBIDDEN then
                  Increment BUSY for selected channel
                  Set CURR.CHANNEL to selected channel
                else
                  goto Repeat:
              end if
            else
              Start BACKOFF Timer
            end if
          else
            if ON Timer=1 then
              Set OFF Timer
              Release reserved channel
            end if
            Decrement ON Timer
          end if
        end if
      end if
    else
      if packet delay greater than threshold then
        Drop burst
        Set OFF timer
      else
        Decrement BACKOFF timer
      end if
    end if
  end for
  for all Nodes do
    if CURR.CHANNEL  $\neq$  0 then
      if BUSYi1 for CURR.CHANNEL then
        Increment COLLISIONS
        Set BACKOFF timer
      else
        Decrement ON timer
        if ON timer=0 then
          Set OFF Timer
          Release reserved channel
        else
          Put CURRENT.CHANNEL in FORBIDDEN
        end if
      end if
    end if
  end for
end for
end while
```

This function gives us the throughput, delay, packet generation rate and drop probability. We have compared the performance of our scheme with FSA. We compare the throughput, delay, packet generation rate and drop probability of both these channel access schemes. For these simulations, we take $A = 10$, $B = 8$ and $M = 16$ and the simulation runs for $T_{max} = 10^6$ time slots. We also compare the delay and throughput performance of our scheme for different values of A , B , and M .

3.3.1 Throughput

As we can see from Fig. 3.2, as we increase the number of nodes in the network from 2 to 1024, RFSA outperforms FSA. This graph trend is expected as with increasing load, the number of utilized time frames also increases but after a certain point throughput goes down again as at the higher loads number of collisions also increases. Since in RFSA we are ensuring the availability of a channel by reserving it, the overall throughput is much better than FSA.

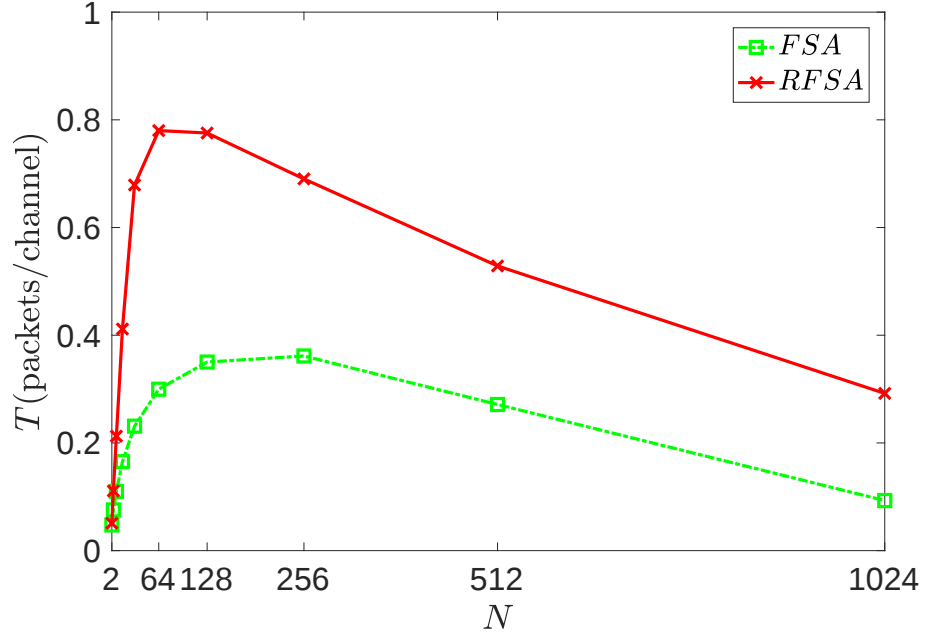


Figure 3.2: Throughput T RFSA vs FSA as function of number of nodes N .

3.3.2 Delay

We can observe in Fig. 3.3 that the delay experienced by each successfully transmitted packet in FSA is slightly less as compared to that in RFSA. This can be understood by the fact that for RFSA a burst gets dropped only if the first packet experiences a delay of more than the mean *OFF* time. However, for FSA if the delay experienced by any packet due to backoff exceeds A , rest of the burst is dropped. For RFSA as lesser collisions result in lesser number of bursts being dropped, it means that last packets in the burst might experience longer delays due to transmission of packets ahead of them. For FSA, since more bursts are dropped so more packets are dropped without experiencing longer delays due to transmission.

The delay for both protocols becomes stable after a point due to the backoff and dropping criteria being used.

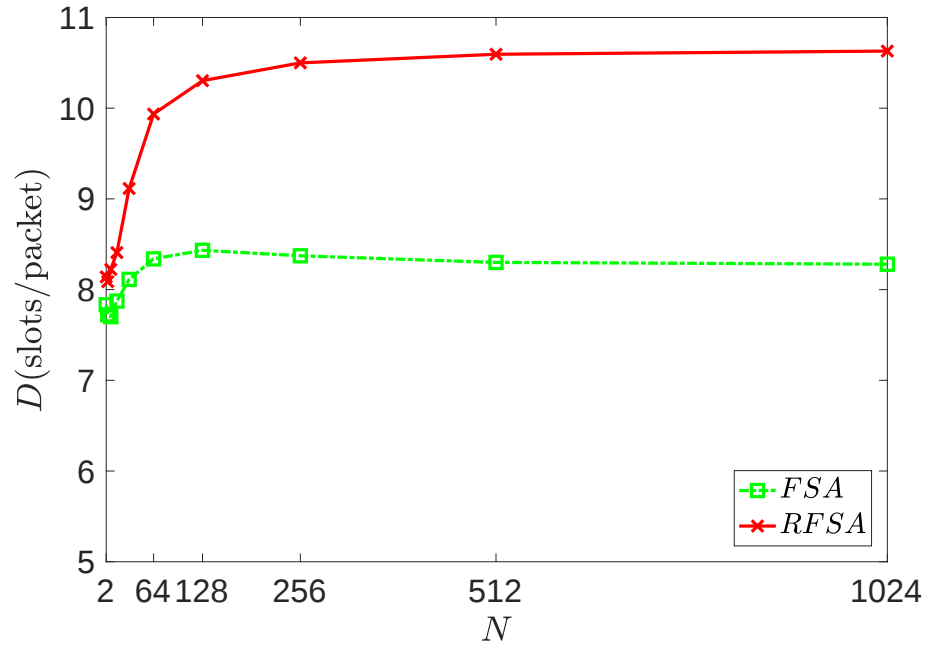


Figure 3.3: Average delay D per successfully transmitted packet RFSA vs FSA as function of number of nodes N .

3.3.3 Energy Consumption

In Fig. 3.4, we can see a plot of retransmissions per packet for both these protocols. For lower loads, the energy consumption of RFSA is much lower as compared to FSA. However, for higher loads, since the number of collisions increases for both the protocols so the number of re-transmissions becomes comparable.

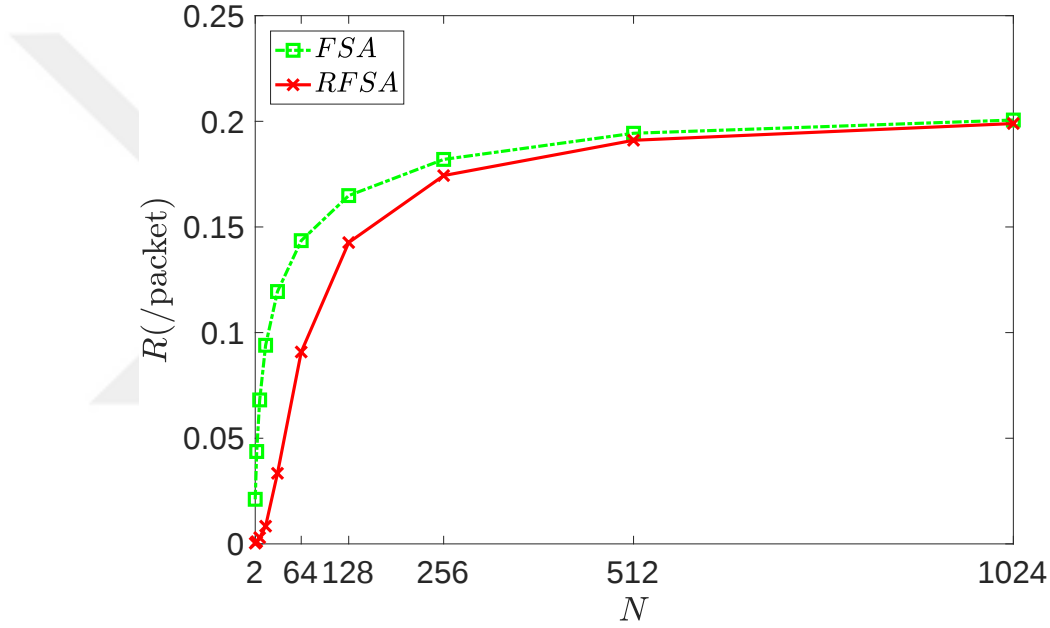


Figure 3.4: Number of retransmissions R per packet RFSA vs FSA plotted as function of number of nodes N .

3.3.4 Packet Drop Probability

As shown in Fig. 3.5, the packet drop probability for RFSA is also much less than FSA. As the delay experienced by packets in RFSA is much less, so the number of packets exceeding the delay threshold and thus being dropped is also much less. For higher load values, the drop probability of both the protocols become comparable as the number of collisions increases at higher loads for RFSA as well.

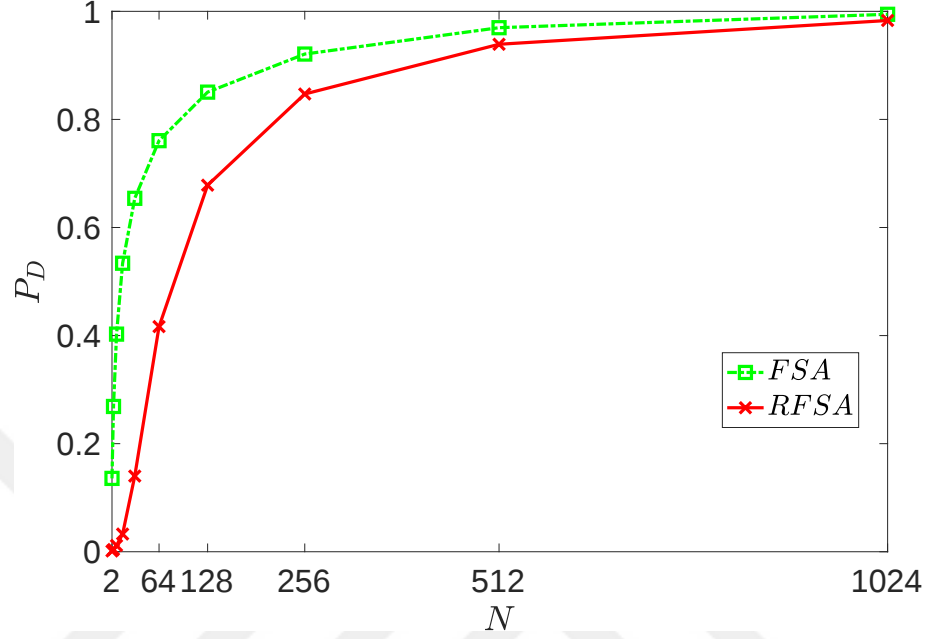


Figure 3.5: Packet Drop Probability P_D RFSA vs FSA as function of number of nodes N .

3.3.5 Delay and Throughput Comparison for varying A , B and M

In this section, we compare the throughput and delay of RFSA for different values of mean ON time, mean OFF time and the number of channels.

First, we compare the throughput and delay of RFSA for $B=8$, $B=16$ and $B=32$. As we increase the value of the mean ON time, throughput performance of our algorithm also increases. As the mean ON time increases, a node captures a channel for a longer period of time, which in turn results in increased channel utilization.

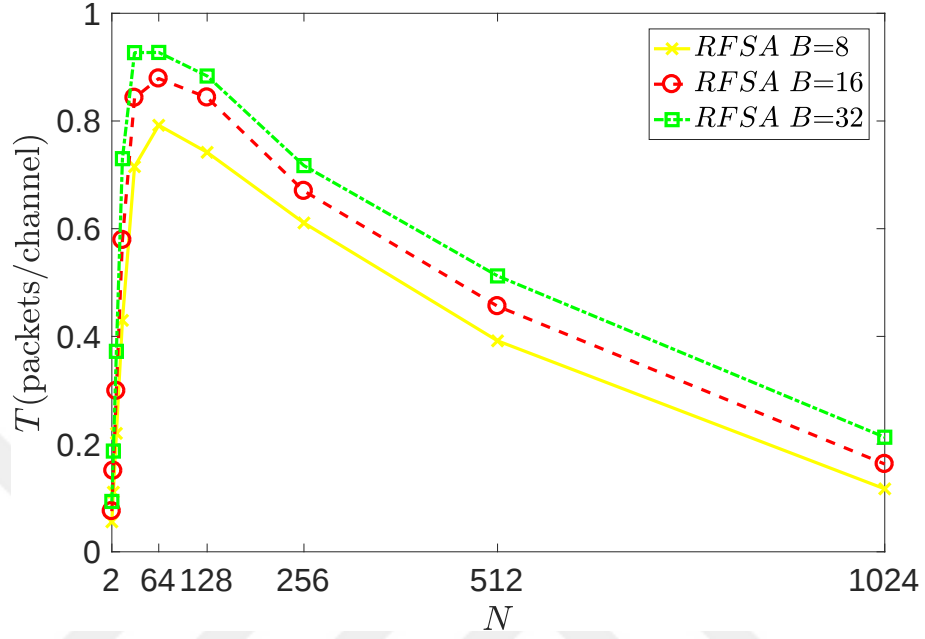


Figure 3.6: Throughput T for RFSA with respect to varying value of mean burst length B .

In terms of delay, however, RFSA performs worse when B increases. This result is also intuitive as increasing the time a channel remains reserved increases the amount of time packets stay in the network waiting to be transmitted. While initially, the delay performances are comparable, the difference becomes much significant for higher traffic loads.

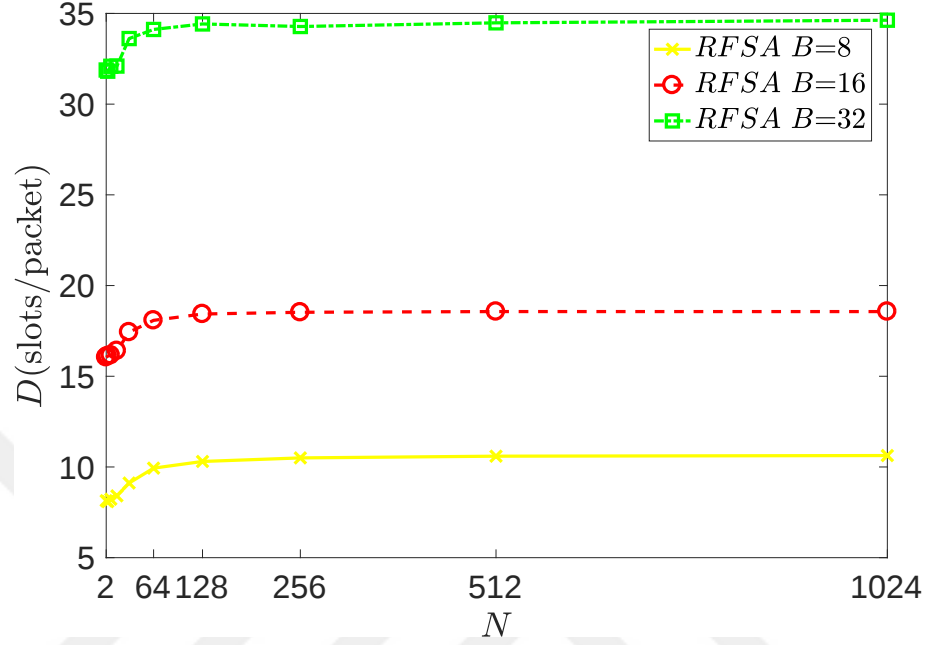


Figure 3.7: RFSA average delay D with respect to varying value of mean burst length B .

We have compared the performance of our protocol for $A = 10$, $A = 20$ and $A = 30$. Both the throughput and delay comparison show us that increasing A has the opposite effect as compared to increasing B in this system model. Increasing A decreases the throughput for less number of nodes as longer sleep times result in less channel utilization. However, as the number of nodes in the network increase, higher values of A mean less load on the system, which results in lesser number of collisions and better throughput.

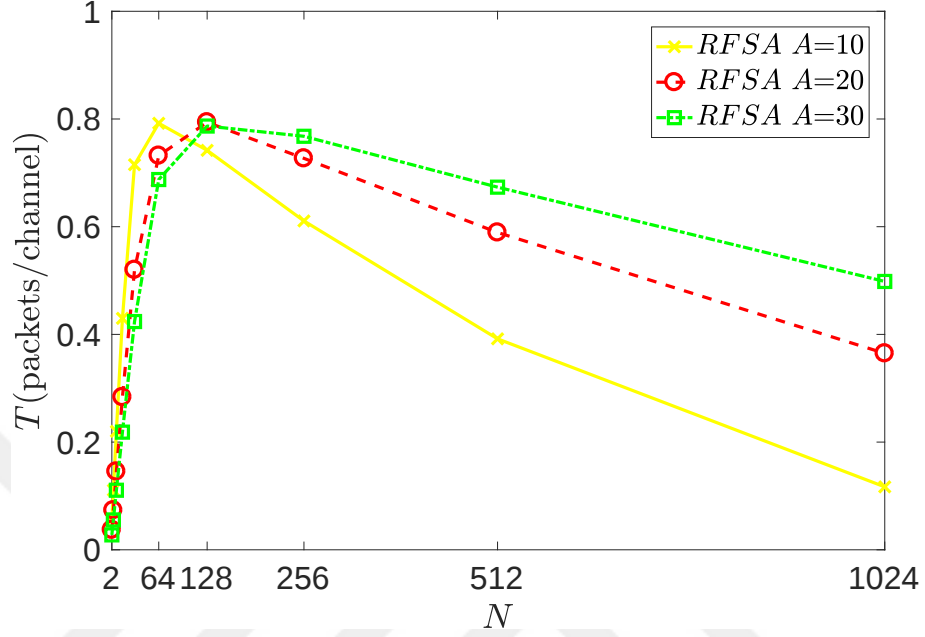


Figure 3.8: Throughput T for RFSA with respect to varying value of mean idle time A .

Delay performance of the protocol decreases by increasing A and the delay becomes higher at higher load values. Longer sleep times result in less traffic load which results in less delay experienced by the first packets in a burst. As this delay is less, so lesser bursts are dropped thus resulting in longer transmission delays experienced by packets. In addition as A determines the upper bound on the delay experienced due to backoff by each packet in the burst, so increasing A also increases the average delay experienced by each successfully transmitted packet.

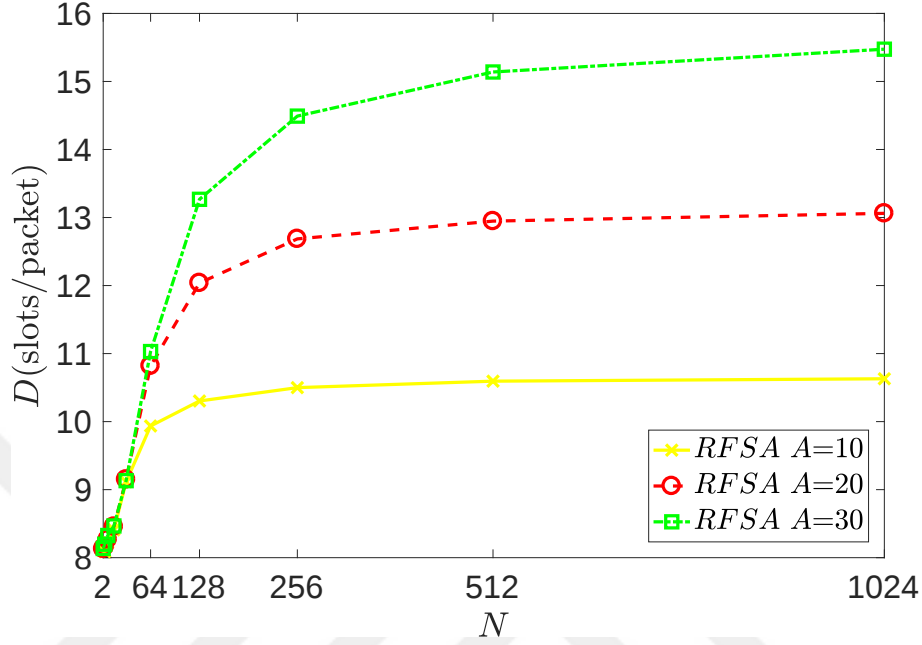


Figure 3.9: RFSA average delay D with respect to varying value of mean idle time A .

In this section, we compare the throughput and delay of RFSA for different values of M . We have fixed $M=16$, $M=32$ and $M=64$ with $A=10$ and $B=8$. The results are given in figures 3.10 and 3.11. Here it is important to mention that we are taking throughput to be the number of successfully utilized frames. As for all three cases, the offered load is the same but the number of frames is higher for $M=32$ and $M=64$, so at lower loads their throughput is also less than $M=16$. But as the load on the system increases, the number of collisions increase for $M=16$ and its performance drops. The other two perform better as the number of available channels is higher and hence the number of collisions is less.

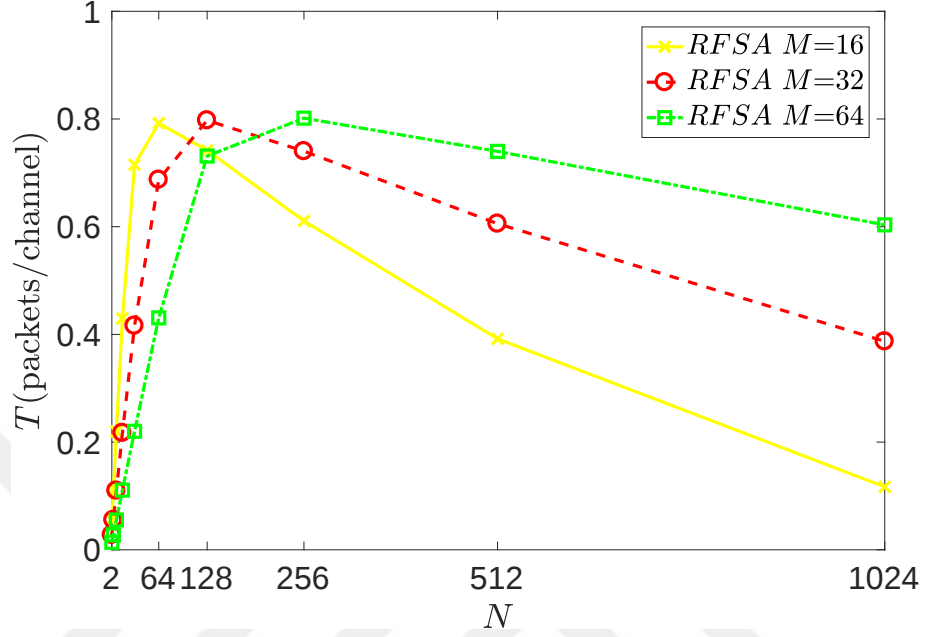


Figure 3.10: Throughput T for RFSA with respect to varying value of number of channels M

In terms of delay, however, RFSA performs better when $M = 32$ and $M = 64$. This result is also intuitive as increasing the number of available channels decreases the number of time packets stay in the network waiting to be transmitted. While initially, the delay performances are comparable, the difference becomes much significant for higher traffic loads.

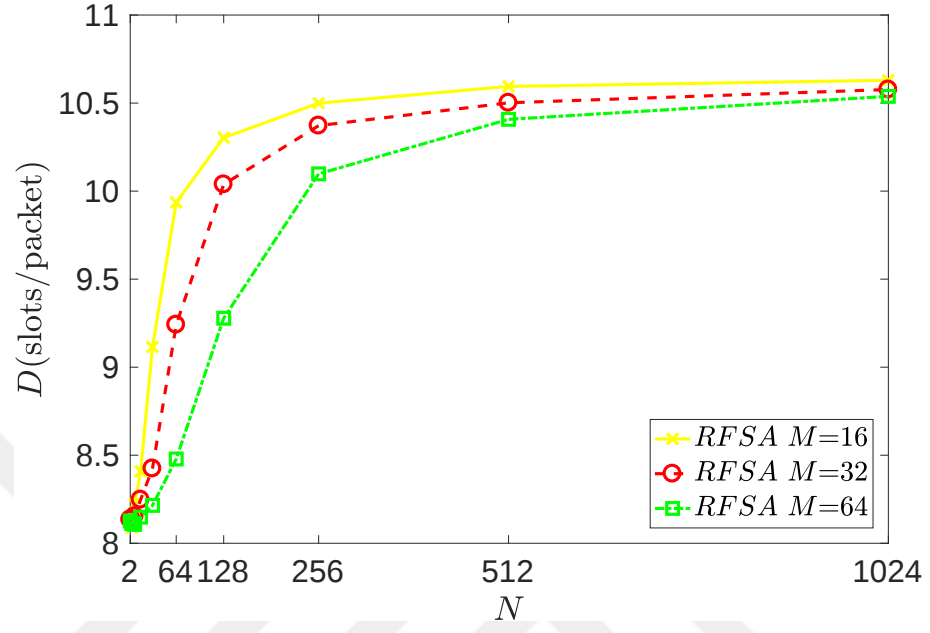


Figure 3.11: RFSA average delay D with respect to varying value of number of channels M .

Chapter 4

Reservation Frame Slotted ALOHA for Multi-Class Networks

In the last chapter, we studied the performance of RFSA for a network in which all nodes belong to a single class. In real-world scenarios, this is often not the case. In the IoT services generally, a variety of devices communicate and work together towards a common task. In such a network, a variety of devices share the channel resources for communicating with the gateway. We can model such a network as a multi-class network in which nodes belong to different classes. In this chapter, we develop a new system model for a more realistic multi-class network. We then try to adapt RFSA for such a network. In a multi-class environment, channel allocation between different classes is an interesting problem for RFSA. We propose three different channel allocation schemes and study their performance as compared to FSA.

4.1 System Model

In this system model, we once again consider a single hop network with one gateway communicating with multiple end nodes. This network is different from the system model from the last chapter as the traffic patterns for all nodes is not the same. We have nodes belonging to three different classes. Two of them are reservation-based classes in which nodes transmit periodically with period P_i and each Class- i is characterized by three parameters:

- P_i : Period of class
- OFF -time distribution in slots with mean $E[OFF_i]$
- ON -time distribution in super-slots with mean $E[ON_i]$

For each reservation-based class, a super-slot is taken to be equal to P_i slots. As we take the mean ON time in super slots, so it is always an integer multiple of period P_i of this class.

For each node, a cycle consists of one ON time followed by an OFF time. The ON time gives us the number of packets sampled by the node in a cycle. For each class, the packets are sampled with a period of P_i . We assume that a single packet is transmitted in a single slot. The OFF time, on the other hand, specifies the number of slots this node remains idle. For each ON time a node attempts to capture a slot for transmission of the first packet. Once a slot is captured, it is reserved for the node with period P_i . If the node is unable to capture a slot due to collision it keeps retrying for P_i slots. If it is unable to capture a slot in P_i attempts, the packet is dropped and it performs contention for the next sampled packet.

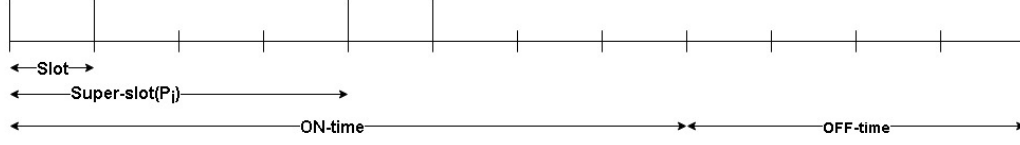


Figure 4.1: System Model

For each class the packet generation rate for each node R_i is defined as the number of packets generated per slot. For the reservation-based classes it can be written as:

$$R_i = \frac{E[ON_i]}{E[OFF_i] + P_i E[ON_i]} \quad (4.1)$$

By fixing the value of $E[ON_i]$, P_i , and R_i we can find the value of $E[OFF_i]$ for this class. We define N_i as the number of nodes of a class in the network and $N_i R_i$ gives us the number of packets generated per slot per class. If M is the number of channels in a slot the load ρ_i generated by each reservation-based class can then be found as:

$$\rho_i = \frac{N_i R_i}{M} \quad (4.2)$$

The overall load is the sum of loads of all classes and can be written as:

$$\rho = \frac{\sum N_i R_i}{M} \quad (4.3)$$

We have a third non-periodic sporadic class which generates traffic with a probability φ . Nodes belonging to the sporadic class sample packets at a rate φ and do not require reservation. Each node attempts to transmit this packet and if transmission fails, the packet is dropped.

4.2 Performance Metrics:

We use three performance metrics to evaluate the performance of our algorithm namely:

- Throughput T ,
- Blocking probability P_b ,
- Average energy consumption E .

We define the throughput T as the number of successful transmissions per channel.

Blocking probability P_b is the average number of packets dropped per generated packet.

We assume that for each node E_{tr} is the energy consumed for transmitting each packet and R is the number of retransmissions per packet generated in the network. We take the average energy consumption of the network to be $E = E_{tr} \times R$.

Since the energy consumption is directly related to the number of retransmissions per packet we take R to be our performance metric of interest.

4.3 Algorithms:

For our study, we compare the performance of four different medium access schemes.

- Frame slotted Aloha (FSA)
- Reservation frame slotted Aloha-Static Partitioning (RFSA-SP)

- Reservation frame slotted Aloha-Static Partitioning with Sharing (RFSA-SPS)
- Reservation frame slotted Aloha-Dynamic Partitioning (RFSA-DP)

All these schemes are used for the multi-class system model as described above. Frame slotted Aloha is used as the performance benchmark as we are offering an improvement to it.

4.3.1 Reservation Frame Slotted ALOHA-SP

In RFSA-SP, we statically partition the channels for each reservation-based class. The channels reserved for one reservation-based class cannot be used by the other reservation-based class. The sporadic class selects uniformly between all the channels as nodes in this class do not require a reservation. Users of each reservation-based class choose uniformly between different slots that are available to this specific class. We define C_1 and C_2 as the number of channels reserved for each class. The channel selection algorithm used by nodes of each reservation-based

class is as follows:

Algorithm 3: Channel selection RFSA-SP

```
if all channels not busy then
    Repeat:
    if Node belongs to Class 1 then
        Randomly select channel from first C1 channels
    else
        Randomly select channel from last C2 channels
    end if
    if Channel not FORBIDDEN then
        Increment BUSY for selected channel
        Set CURR_CHANNEL to selected channel
    else
        goto Repeat:
    end if
else
    Decrement ON timer
end if
```

4.3.2 Reservation Frame Slotted ALOHA-SPS

In RFSA-SPS, we again statically partition the channels for each class but we also let the classes share some channels. Each node first tries to capture a channel reserved for its own class. If all such channels are in use, it then tries to capture one of the shared channels. For the reservation-based class if the current channel belongs to its own class it is reserved for future use however if it belongs to the shared channels it performs contention for the next packet. The sporadic class can still select any channel regardless of whether it's reserved for a class or shared. The channel selection algorithm used by each reservation-based class is

given below:

Algorithm 4: Channel Selection RFSA-SPS

```
if all channels for this class not busy then
  Repeat:
    Randomly select channel from channels reserved for class
    if Channel not FORBIDDEN then
      Increment BUSY for selected channel
      Set CURR_CHANNEL to selected channel
    else
      goto Repeat:
    end if
else
  Repeat1:
    Randomly select channel from shared channels
    if Channel not FORBIDDEN then
      Increment BUSY for selected channel
      Set CURR_CHANNEL to selected channel
    else
      goto Repeat1:
    end if
end if
```

4.3.3 Reservation Frame Slotted ALOHA-DP

In RFSA-DP channels are dynamically partitioned between the classes. If a node manages to secure a channel, it is reserved for this class and nodes from the other class cannot access this channel. Reservation of a channel ends when all nodes of this class utilizing this channel complete their transmission. So the partitioning of resources happens dynamically based on the traffic pattern. The channel selection algorithm utilized by nodes of reservation-based classes is as follows:

Algorithm 5: Channel Selection RFSA-DP

```
if all channels not busy then
  Repeat:
  Randomly select channel
  if Channel not FORBIDDEN and not reserved for other class then
    Increment BUSY for selected channel
    Set CURR_CHANNEL to selected channel
  else
    goto Repeat:
  end if
else
  Decrement  $ON$  timer
end if
```

The gateway in case of two reservation-based classes provides two bits of information about a slot to all nodes:

- 00: The slot is available to all classes
- 01: Slot is available to nodes of class 1
- 10: Slot is available to nodes of class 2
- 11: Slot unavailable

So these algorithms can easily be implemented with two-bit feedback provided by the gateway.

4.4 Simulation Setup:

For our simulations we take two classes with periods P_1 and P_2 . We fix the packet generation rate for each class to be $R_1 = R_2 = R = 0.01$. Our third sporadic class generates traffic with a probability $\varphi = 0.01$. We take the ratio of the periods of

the reservation-based classes to be $R_p = \frac{P_2}{P_1} = 16$. N is the total number of nodes in the network and there is an equal number of nodes of each class in the network i.e. $N_1 = N_2 = N_3 = N/3$. Since for this system model, we know the exact load on the network so we plot our performance metrics versus ρ . To find the number of nodes in the network for each value of ρ we put these values in Equation 4.2:

$$N_2 = \frac{\rho M}{R} \quad (4.4)$$

For this simulation, we vary ρ from 0.2 to 4. The parameters used for these simulations are as follows:

P_1	2
P_2	32
R	0.01
M	16
$E[ON_i]$	10
T_{max}	10^7

For each reservation-based class we fixed the $E[ON]$ to be 10. Putting the value of P, R , and $E[ON]$ in Equation 4.1 we can find the $E[OFF_i]$ for each class.

4.5 Simulation Results:

4.5.1 Reservation Frame Slotted ALOHA-SP

In this subsection, we study the results of RFSA with static partitioning of channels among the classes. As discussed earlier, we have three classes with an equal packet generation rate of 0.01. All three classes have an equal number of nodes in the network and we vary the load in the network by changing the total number of nodes. For the reservation-based classes, we allocate an equal number of channels i.e., half of the total number of channels. Nodes from each reservation-based class

only use the channels allocated to their class. The nodes from the sporadic class are, however, free to select any channel for transmission as long as it is available in the current slot.

First, we study the throughput of RFSA-SP as compared to FSA in this multi-class network. As we can see from Fig. 4.2, RFSA-SP outperforms FSA in terms of throughput. The maximum throughput achieved by the protocol as well as the throughput at high traffic loads are better than FSA. An important point to note in this figure is that beyond $\rho = 2.6$, the throughput performance of RFSA-SP drops at a much higher rate. For understanding this, we must keep into account that one-third of the traffic for this system model does not benefit from the reservation, so at the higher loads performance of the algorithm drops. In addition, as at higher loads, a higher number of nodes try to access this small subset of channels, the number of collisions increases rapidly.

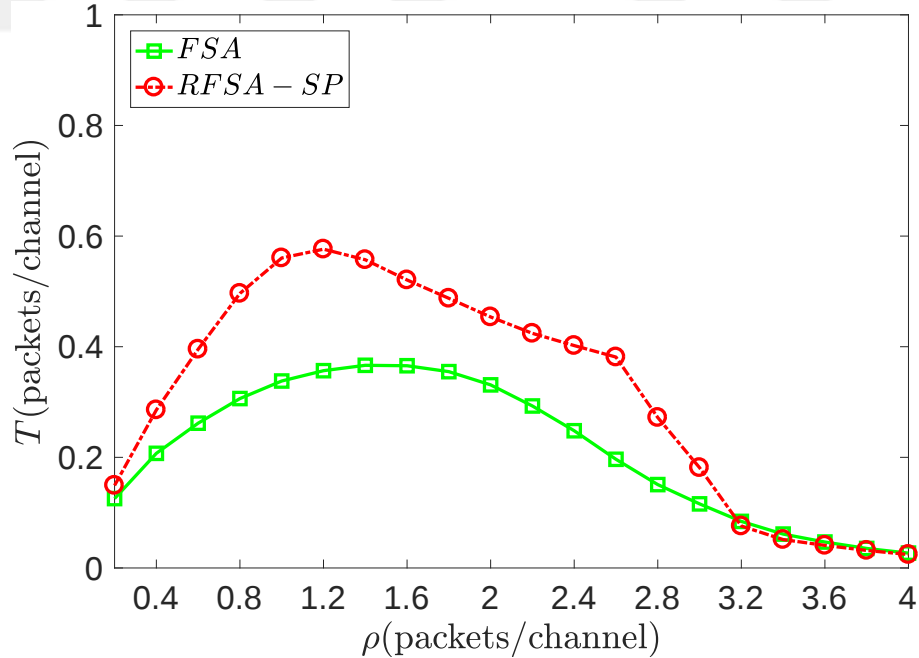


Figure 4.2: Throughput T as a function of load ρ for FSA and RFSA-SP.

Next, we take a look at the blocking probability for both these protocols. As we expected the blocking probability for RFSA-SP is much less than FSA. For this system model, a burst is only dropped if the delay exceeds the period of

the class. As for RFSA, the number of collisions is much less than FSA, so fewer nodes experience a delay of more than P for their first packet. Another important factor is that for FSA nodes perform contention for each packet in the burst so the probability of experiencing a delay of more than P in a burst is higher. The blocking probability for RFSA-SP increases rapidly for $\rho > 2.6$, as for high load as a number of collisions is higher. Since nodes of the sporadic class drop their packets upon the first collision and do not try retransmissions these packets contribute significantly to the increase in the blocking probability at higher loads.

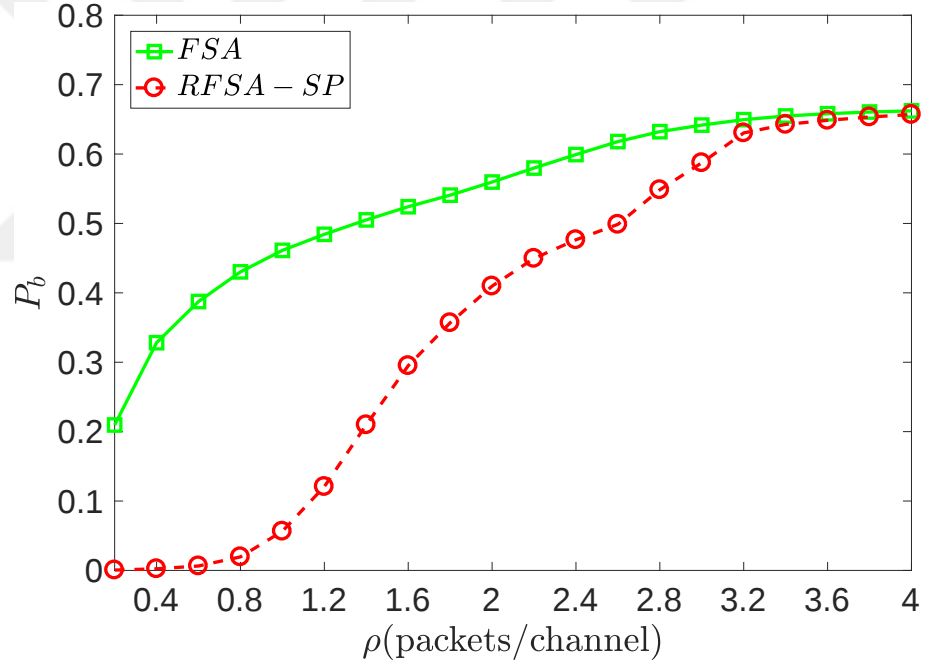


Figure 4.3: Blocking probability P_b as a function of load ρ for FSA and RFSA-SP.

In Fig. 4.4, we compare the energy performance of FSA and RFSA-SP. For lower loads the energy performance of FSA and RFSA-SP is comparable but at higher loads, we can see a significant increase in the energy consumption of FSA. For $\rho > 2.6$, the energy consumption of RFSA-SP also increases rapidly.

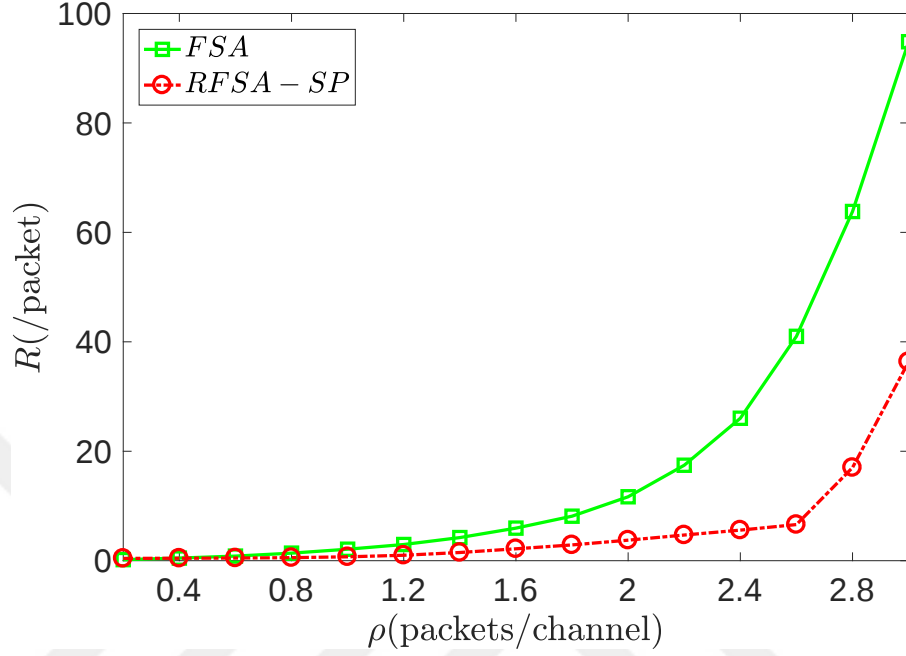


Figure 4.4: Number of retransmissions R per packet as a function of load ρ for FSA and RFSA-SP.

4.5.2 Reservation Frame Slotted ALOHA-SPS

In this subsection, we discuss the performance of RFSA-SPS as compared to FSA as well as RFSA-SP. We try to evaluate the effect of having shared channels in our network. For these simulations we have fixed $M=16$. So we fix 4 and 8 channels as shared in our network and study the performance of RFSA-SPS for these shared channels to gain some insight about the effect of sharing on performance. In the shared channels, nodes from all classes are allowed to transmit their packets, but no reservations for the future slots can be made. So the reservation-based classes need to contend for each packet they transmit. For this section, we mention the number of shared channels as follows: For example RFSA-SPS₄ represents RFSA-SPS with 4 shared channels.

First, we study the throughput performance of RFSA-SPS₄ as compared to FSA. For RFSA-SPS₄ the throughput graph has a similar trend as RFSA-SP.

For $\rho > 2.2$ we see a significant drop in throughput but after the initial drop throughput remains stable. So RFSA-SPS₄ performs better than FSA for low and high load values, but for a few values in the middle, its throughput is less than FSA.

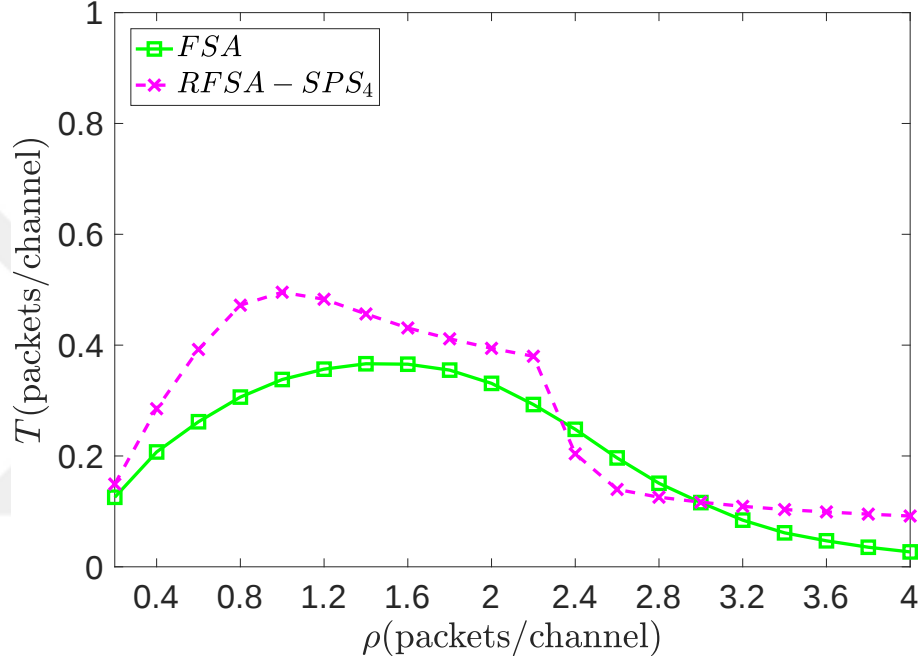


Figure 4.5: Throughput T as a function of load ρ for FSA and RFSA-SPS₄.

In Fig. 4.6 we have the blocking probability plot for RFSA-SPS₄ and FSA. The blocking probability plot follows a similar pattern to the throughput plot for these two protocols. At lower values of ρ , RFSA-SPS₄ has a much less blocking probability as compared to FSA. For some intermediate values of ρ FSA outperforms RFSA-SPS₄ but at higher loads once again RFSA-SPS₄ has lower blocking probability.

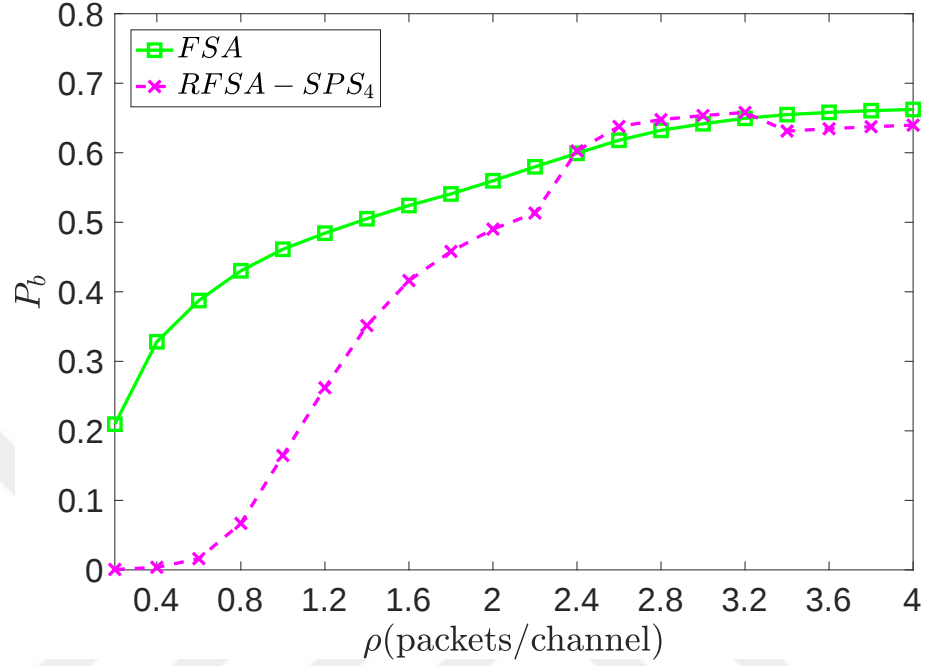


Figure 4.6: Blocking probability P_b as a function of load ρ for FSA and RFSA-SPS₄.

Next, we analyze the average energy consumption of RFSA-SPS₄ as compared to FSA. At lower load values the energy consumption of both the protocols is comparable. For a few load values RFSA-SPS₄ consumes more energy as compared to FSA but at higher loads, its energy consumption is much less than FSA. This graph trend is similar to the throughput graph for these protocols.

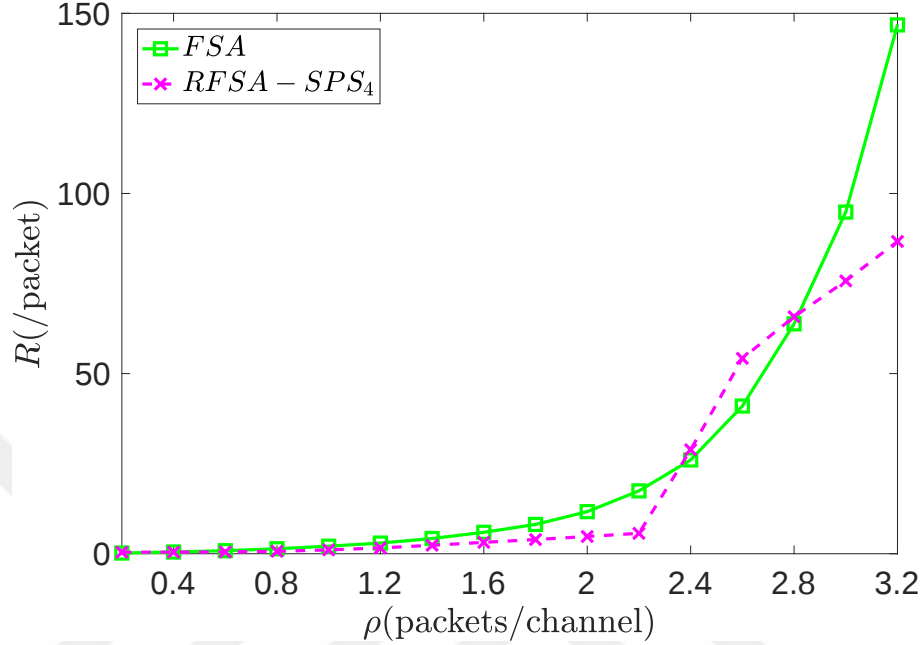


Figure 4.7: Number of retransmissions R per packet as a function of load ρ for FSA and RFSA-SPS₄.

The behavior of RFSA-SPS₄ can be easily understood. At lower load values, the number of collisions is low, resulting in fewer retransmissions and better throughput. When load increases, we observe a sudden drop in throughput and increase in collisions and retransmissions. This sudden drop happens when the number of available fixed channels for each class is low and multiple nodes are contending to access these channels, hence resulting in collisions. Once all fixed channels are reserved, nodes now access the shared channels hence stabilizing the number of collisions and throughput of the system.

Now we compare the throughput of RFSA-SPS₈ with FSA. As we increase the number of shared channels we observe a drop in the overall throughput. This graph also follows the same trend as RFSA-SP and its throughput drops beyond $\rho = 1.6$. So as we add shared channels in our system and decrease the number of fixed channels, the throughput drop occurs at lower load but the drop is less. And throughput remains stable after this drop. As compared to FSA the throughput of RFSA-SPS₈ is better for really low and high loads.

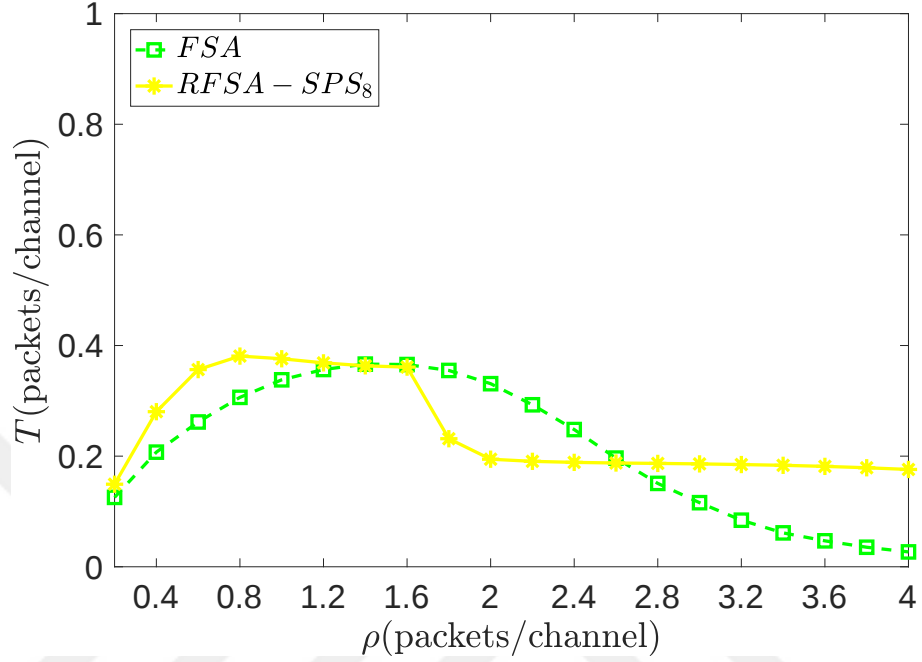


Figure 4.8: Throughput T as a function of load ρ for FSA and RFSA-SPS₈.

In Fig. 4.9, the Blocking probability of both these schemes is plotted. This plot also follows the same pattern as the throughput plot. At low loads, RFSA-SPS₈ has a much less blocking probability than FSA. As the throughput drops at $\rho = 1.6$, the blocking probability increases and becomes higher than FSA. At much higher loads, however, the blocking probability of FSA is again higher.

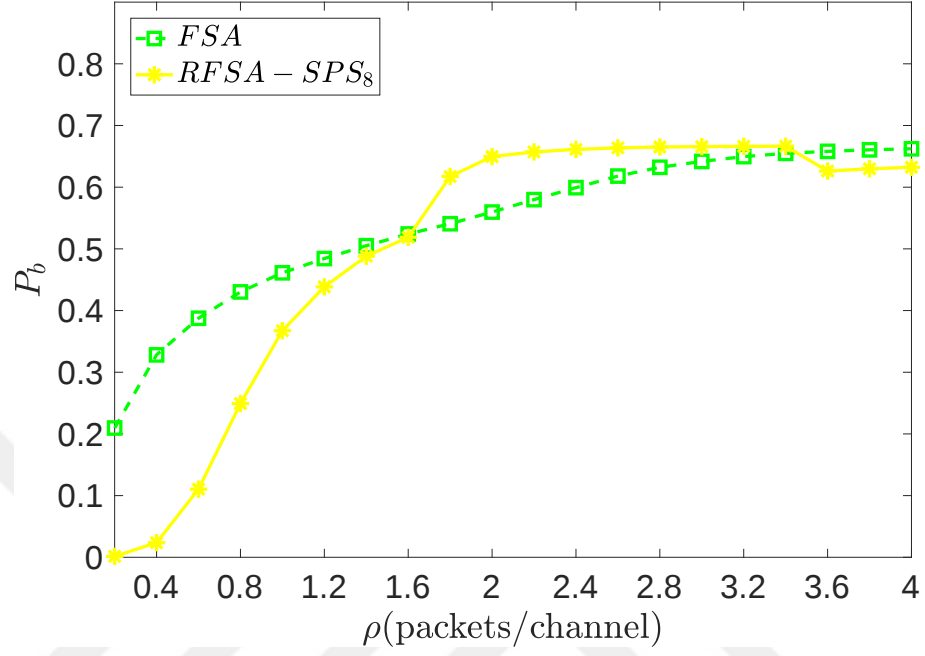


Figure 4.9: Blocking probability P_b as a function of load ρ for FSA and RFSA-SPS₈.

Energy consumption for these schemes has a similar trend with RFSA-SPS₈ consuming less energy at lower and higher loads but for some intermediate values FSA outperforms.

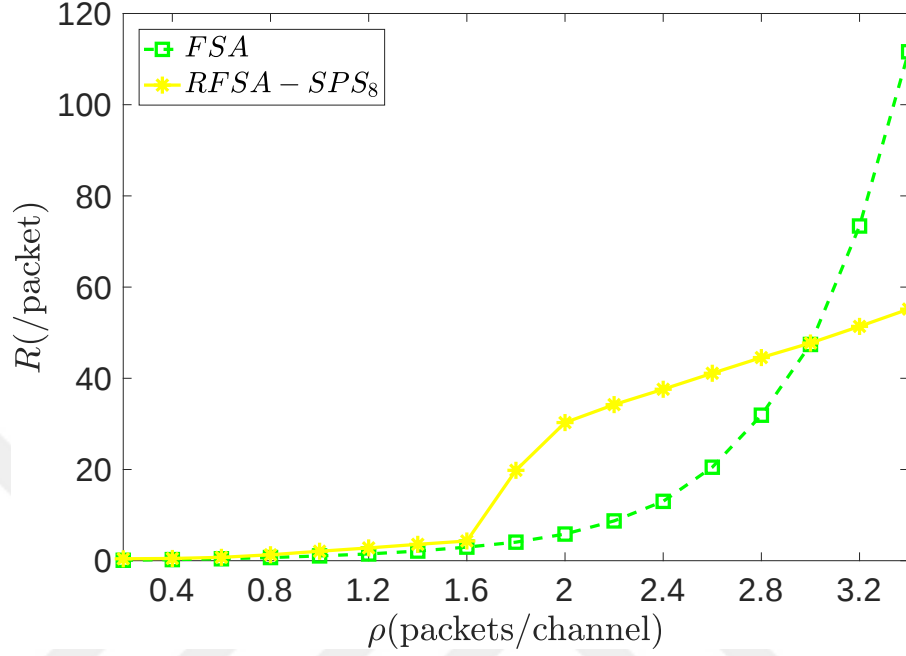


Figure 4.10: Number of retransmissions R per packet as a function of load ρ for FSA and RFSA-SPS₈.

As we increase the number of shared channels the overall throughput drops but the performance at higher loads improves.

4.5.3 Reservation Frame Slotted ALOHA-DP

In this section, we compare the performance of RFSA with Dynamic Partitioning as compared to FSA. In this algorithm, all three classes randomly select a channel from all the available channels. If a node from a reservation-based class manages to capture a channel, this channel is reserved for the nodes of the same class and nodes of the other reservation-based class cannot access it. The nodes of the non-periodic class, however, have the freedom to access any channel as long as it is not in use in the current slot. In this way, nodes can select a channel for transmission if it is either reserved for their class or not in use by any class. When all nodes of one class complete their transmission over a channel, that channel is freed and can now be used by any node.

First, we compare the throughput of RFSA-DP and FSA. As we can see from Fig. 4.11, RFSA-DP does outperform FSA at both low and high load values. An important observation here is that as opposed to the static partitioning schemes there are no sudden throughput drops for this algorithm. Dynamic nature of this algorithm allows it to adapt to the traffic and hence provide a stable throughput.

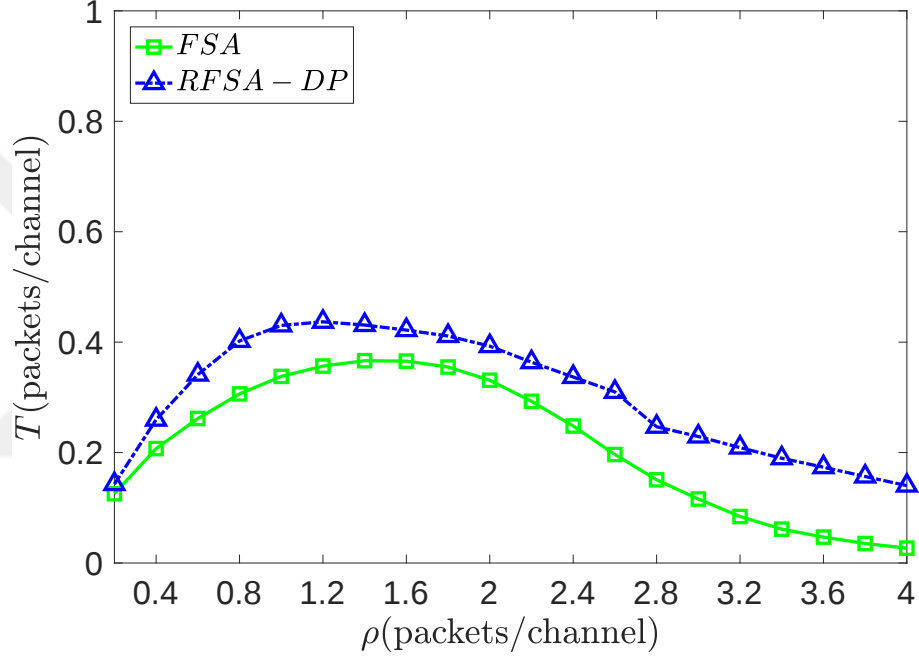


Figure 4.11: Throughput T as a function of load ρ for FSA and RFSA-DP.

Blocking probability plot shows us that RFSA-DP has less blocking probability than FSA at all load values. While there is a significant difference between the blocking probabilities of the two protocols at lower load values, this difference reduces at higher load values.

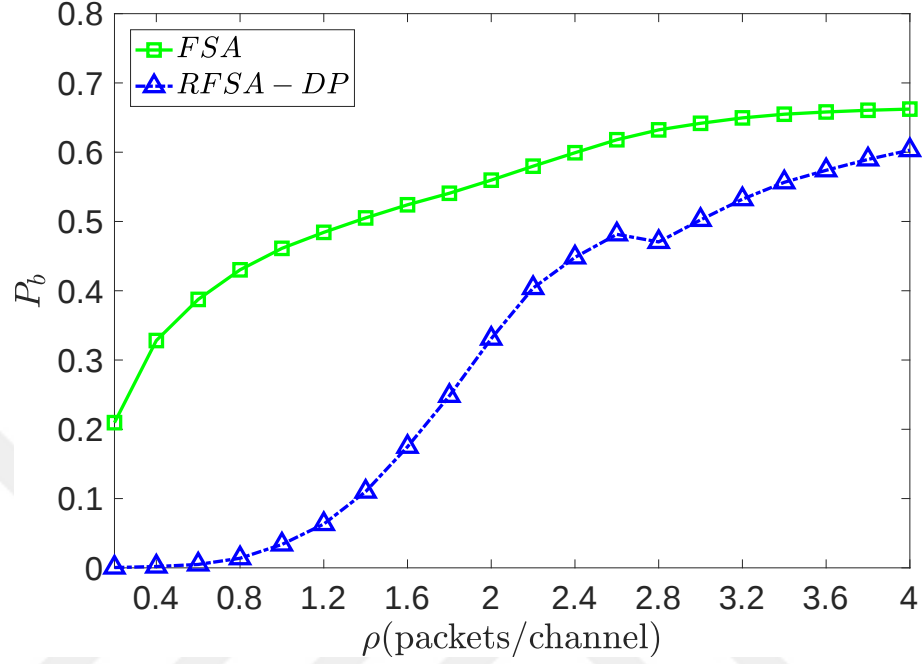


Figure 4.12: Blocking probability P_b as a function of load ρ for FSA and RFSA-DP.

In Fig. 4.13, we compare the energy performance of FSA and RFSA-DP. For lower loads the energy performance of FSA and RFSA-DP is comparable but at higher loads, we can see a significant increase different between the energy consumption of the two protocols. The number of collisions is less for RFSA-DP, which results in a lesser number of retransmissions.

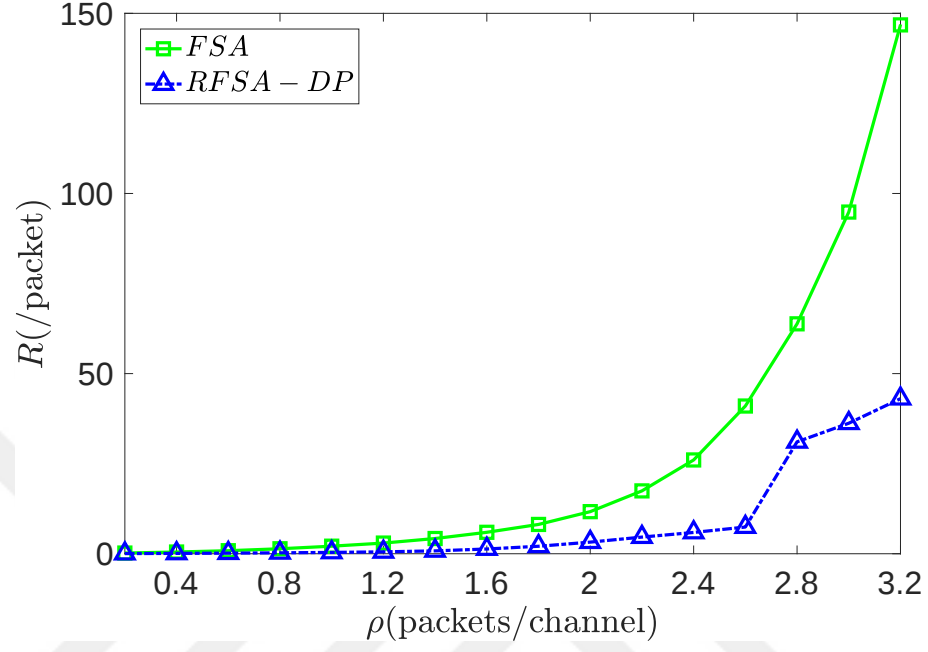


Figure 4.13: Number of retransmissions R per packet as a function of load ρ for FSA and RFSA-DP.

4.5.4 Static and Dynamic Partitioning Throughput Comparison

We compare the throughput of RFSA-SP, RFSA-SPS, and RFSA-DP to gain some insights about their behavior.

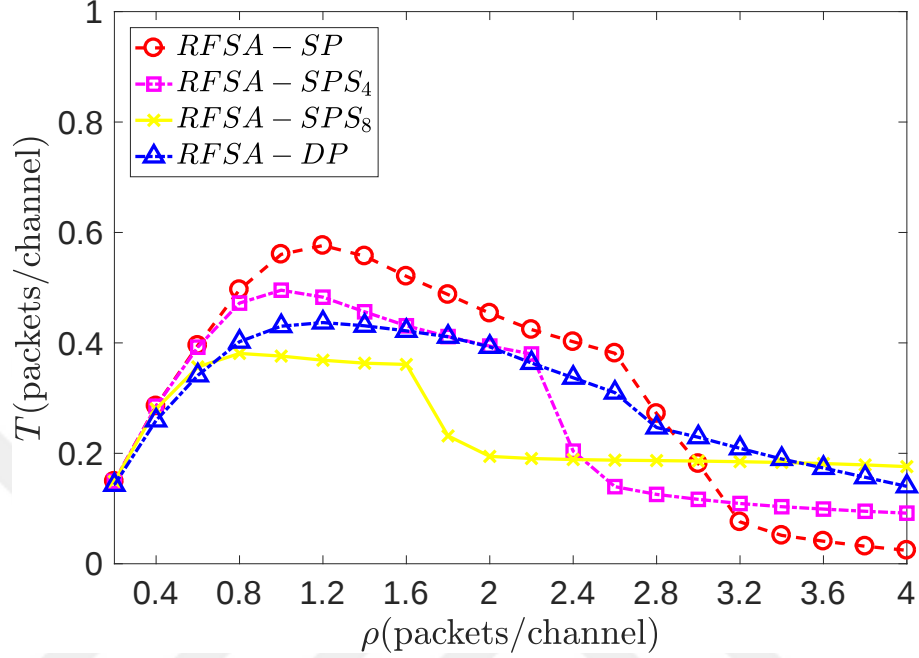


Figure 4.14: Throughput T as a function of load ρ for RFSA-SP, RFSA-SPS and RFSA-DP.

In Fig. 4.14, we can observe that RFSA-SP outperforms RFSA-SPS and RFSA-DP in terms of maximum throughput achieved. Adding shared channels has a negative effect on the maximum achievable throughput by the scheme. But for $\rho > 2.6$, RFSA-SP gets outperformed by RFSA-SPS. For high load values RFSA-SPS has better performance and adding shared channels has a positive impact on this performance. Similarly, Dynamic Partitioning has lower throughput at low loads as compared to RFSA-SP and RFSA-SPS but outperforms these schemes at higher load values. It also manages to maintain a stable throughput with no sudden drops in the throughput.

If we try to understand the behavior of these protocols, we can understand that fixing channels for different classes gives us good performance but beyond a certain load, the throughput crashes. This can happen because at high loads most of these fixed channels are in use and new nodes contending get locked out of the system due to collisions on the few available channels. When we add shared channels in the system, we reduce the number of fixed channels hence

reducing the throughput at low loads due to a lesser number of nodes reserving channels. When all the fixed channels are reserved, nodes can access the shared channels. Having this option results in better throughput at higher loads and the throughput drop being less as compared to RFSA-SP. But as the number of fixed channels is less so the throughput drop happens at lower load values as compared to RFSA-SP.

From this comparison, we can conclude that RFSA-SP can be useful for networks with low traffic loads but for networks with high traffic loads RFSA-SPS and RFSA-DP are better candidates. An important point to understand here is that in this system model the traffic load per class is fixed, but this may not always be the case. In cases where the traffic loads for classes are fixed static partitioning is implementable, but for networks where traffic load per class is not fixed dynamic partitioning might be the only solution.

4.5.5 Static and Dynamic Partitioning with Unequal Channel Allocation

In heterogeneous networks with multiple classes, sometimes traffic from one class is given preference over the other class. In this section, we study the effect of unequal channel allocation on the throughput of our static and dynamic partitioning protocols.

For the case of RFSA-SP, we can incorporate unequal channel distribution by modifying the number of channels reserved for each class. So instead of allocating an equal number of channels for each class, we reserve $3/4$ th of the channel resources for Class-1 and $1/4$ th of the resources for Class-2. Since we are using number of channels $M = 16$, so we reserve 12 channels for Class-1 and 4 channels for Class-2. Now we compare the throughput for RFSA-SP with unequal channel distribution (RFSA-SPU) to the throughput we obtained in the previous section for RFSA-SP.

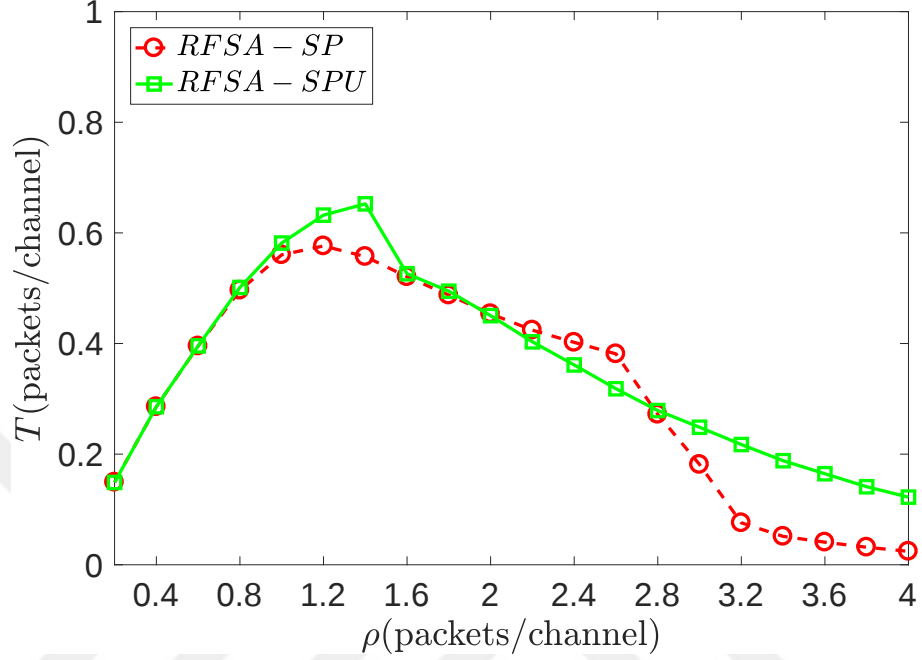


Figure 4.15: Throughput T as a function of load ρ for RFSA-SP and RFSA-SPU.

So we see an improvement in the overall throughput of the protocol. The throughput drop is significantly smaller and the throughput performance at high load values is much improved. All these can be understood by the fact that there is a significant difference between the periods of the two classes. The channels reserved for Class-1 can only be utilized by two nodes of this class at a time, however, the duration a node uses a channel is small. For the case of Class-2, a single channel can be simultaneously used by 32 nodes at a time ideally. So reserving more channels for nodes of Class-1, allows more nodes of this class to transmit and hence improves the throughput. So the ratio of the periods of two classes can play an important role in determining the throughput of the protocol.

For RFSA-DP one way of incorporating unequal channel distribution can be limiting the number of channels available for reservation to one class. We want Class-1 to be able to access double the number of channels as compared to Class-2. So for this scenario we can allow Class-1 to dynamically reserve channels from all the 16 channels and limit the set of channels available to Class-2 to 8. With this setup, we plot the throughput of RFSA-DP with unequal channel distribution

(RFSA-DPU) and compare it with the throughput of RFSA-DP.

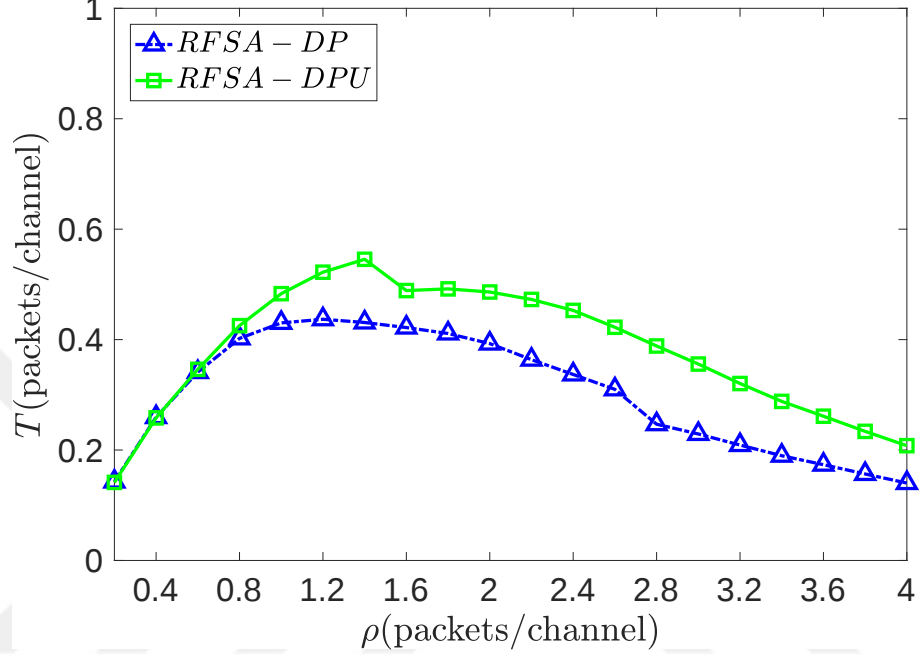


Figure 4.16: Throughput T as a function of load ρ for RFSA-DP and RFSA-DPU.

In this case, as well the throughput of the protocol improves as we allow Class-2 to access fewer channels as compared to Class-1. This is again because of the fact that nodes from Class-2 tend to capture a channel for a longer period of time as compared to Class-1.

4.5.6 Performance of Static and Dynamic Partitioning for varying traffic patterns

In the simulation results discussed above we consider a uniform traffic pattern in which nodes from each class generate packets with an equal rate and there is an equal number of nodes for each class. As we observed from these results static partitioning provides us the best throughput for such a network. As for static partitioning the number of channels allocated to each class are fixed, so prior knowledge about the packet generation rate and number of nodes of each class is

required to benefit from this channel allocation scheme. However, traffic patterns in IoT networks may not always be predictable. In this section we analyze the performance of RFSA-SP and RFSA-DP for varying traffic patterns. We again consider three different classes. The first and second classes are reservation-based and the third class is sporadic.

We define the individual loads from each class as ρ_1, ρ_2 and ρ_3 . The system parameters for these simulation results are as follows

$$\rho_1 = \alpha \times \rho_2 \quad (4.5)$$

We plot the throughput of the two channel allocation schemes as a function of α for three different values of ρ_3 . The simulation parameters for these results are as follows:

P_1	2
P_2	4
R	0.1
M	16
$E[ON]_i$	10

We take these results for $\rho_3 = \rho/2, \rho_3 = \rho/3$ and $\rho_3 = \rho/4$. For RFSA-SP we equally partition the channels between first and second class.

In Figure 4.17 we observe that for $\rho_3 = \rho/2$ the throughput performance of RFSA-DP is more stable as compared to RFSA-SP. For high and low values of α RFSA-DP outperforms RFSA-SP.

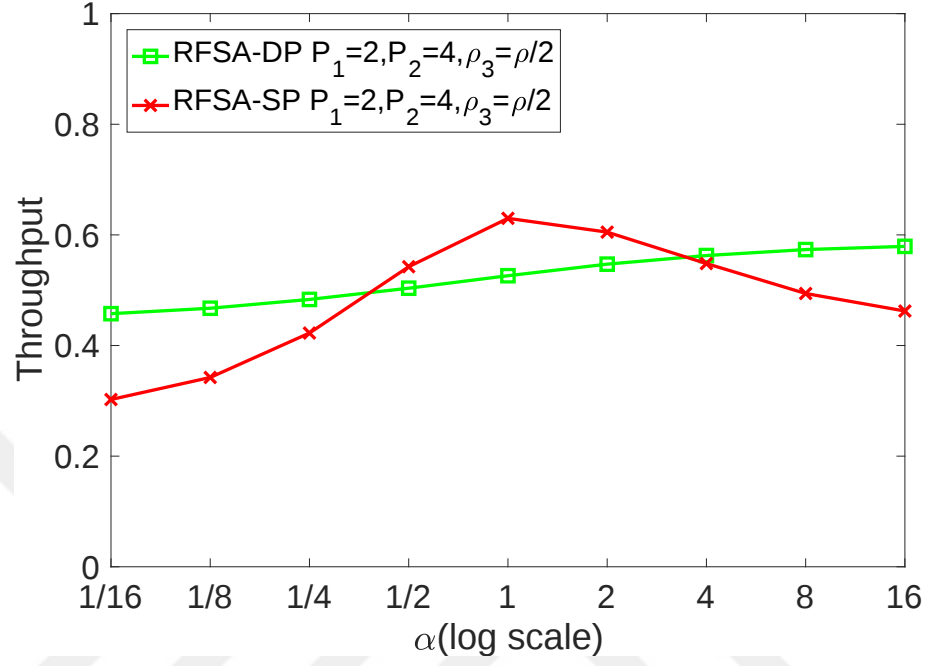


Figure 4.17: RFSA-SP and RFSA-DP throughput T as a function of α for $\rho_3 = 1/2$

We see a similar trend for $\rho_3 = \rho/3$, however, for this case we see a slight drop in the overall throughput of the protocols. This drop is due to the fact that there are lesser number of nodes from class 3 generating sporadic traffic.

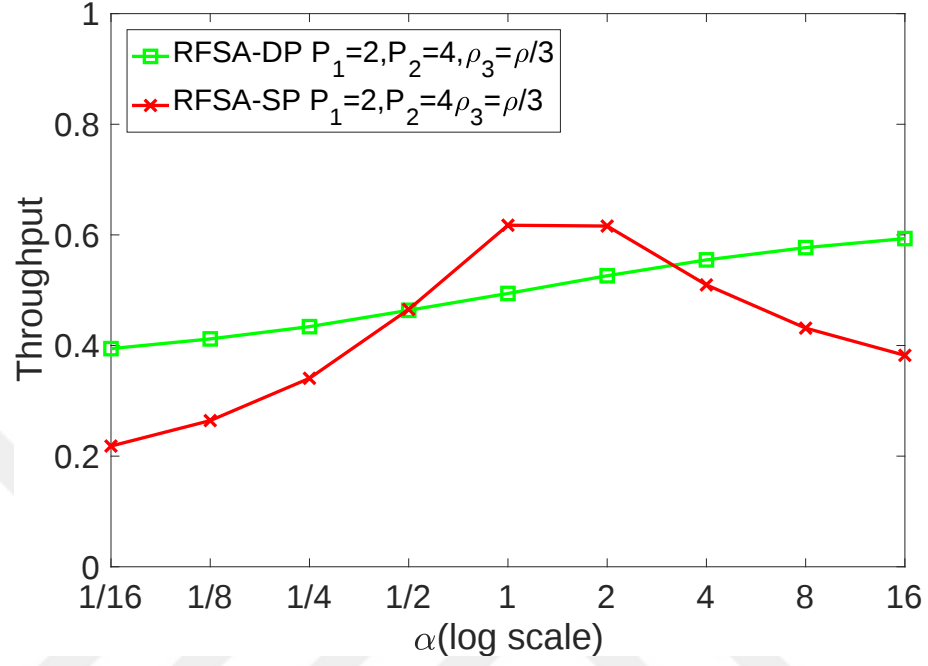


Figure 4.18: RFSA-SP and RFSA-DP throughput T as a function of α for $\rho_3 = 1/3$

For $\rho_3 = \rho/4$ only one fourth of the overall load ρ is contributed by the third class. The overall throughput for the schemes is once again slightly reduced, however, the trend remains same.

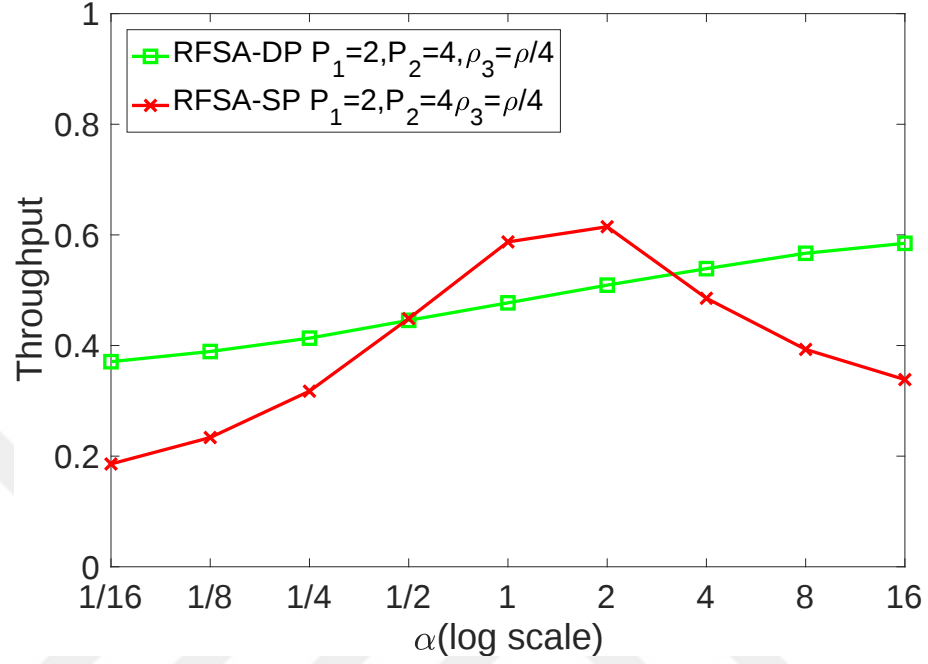


Figure 4.19: RFSA-SP and RFSA-DP throughput T as a function of α for $\rho_3 = 1/4$

We can also observe from this graph that as we decrease the value of ρ_3 , the difference between the performance of these two schemes increases. This is due to the fact that nodes of Class-3 can equally access all the channels in both these schemes.

From these results we can conclude that RFSA-DP is more robust to changes in traffic patterns as compared to RFSA-SP. For taking advantage of RFSA-SP to full we need to ensure that the packet generation rates and loads for each class are fixed, thus allowing us to allocate channels accordingly beforehand. For networks where traffic patterns are constantly changing RFSA-DP performs robustly by offering a stable throughput.

Chapter 5

Conclusions

In this thesis, we studied the performance of reservation frame slotted Aloha for IoT networks with alternating busy and idle periods. We offer an adaptation of this scheme for a heterogeneous IoT network.

First, a homogeneous network model is presented in which all nodes belong to the same class and generate data bursts with a unity period during their busy period. The load on the network in this system model is not fixed as it depends on the amount of time required for transmission of a data burst. The performance of RFSA as compared to FSA for this system model is studied and it is observed that RFSA significantly outperforms FSA in terms of throughput, energy consumption and blocking probability. As reserving the channel for a node once it successfully transmits the first packet in the burst, reduces the number of time each node performs contention, the overall number of collisions in the network is shown to be significantly reduced. We also study the effect of changing the mean *ON* time, mean *OFF* time and the number of channels (number of slots in a frame) on the performance of RFSA in this system model.

Based on the insights gained from the homogeneous network model, we presented a heterogeneous system model with three different classes. Two of these classes have alternating busy and idle periods with packets being periodically

sampled during the busy period. The load in this system model is fixed as the per-class ON times are fixed to be integer multiples of per-class periods. A third sporadic class samples packets randomly. For all three classes, the overall rate is fixed and equal. In such a network, we need to allocate channels for each periodic class to be able to use RFSA. We propose three different channel allocation schemes for this model and comparatively study their performance as compared to FSA.

Static partitioning gives the best throughput at low traffic loads, but its throughput drops significantly at high loads. Having shared channels which cannot be reserved by any nodes improves the performance at higher loads but reduces throughput at low loads. As having shared channels reduces the number of channels available for reservation to each class, the throughput is less than static partitioning at low load values. Dynamic channel partitioning allows nodes to reserve channels for their own class once they capture it. Once all nodes transmitting on this channel finish transmission, the channel is released. While the throughput of dynamic channel partitioning is less than static partitioning at low load values, it offers better throughput at high load values. Unlike static partitioning and static partitioning with sharing, there are no sudden drops in throughput as load increases. This is due to the flexibility of the algorithm to handle different traffic loads. Dynamic channel partitioning can be an attractive candidate for networks in which the traffic loads per class are not fixed and traffic patterns change dynamically. Selection of a good channel allocation scheme depends on the traffic model and system parameters.

The problem of adapting RFSA for a multi-class network with more than three classes is left for future work. Other future directions include but are not limited to adding priority between classes, studying fairness in the system, and studying RFSA-DP for a network in which traffic loads are not fixed.

Bibliography

- [1] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A Survey,” *Computer Networks*, vol. 54, no. 7, pp. 2787 – 2805, 2010.
- [2] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of things (IoT): A vision architectural elements and future directions,” *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.
- [3] “IEEE Standard for Local and metropolitan area networks-Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1 : MAC sublayer,” *IEEE Std 802.15.4e-2012*, 2012.
- [4] R. Jurdak, C. V. Lopes, and P. Baldi, “A survey, classification and comparative analysis of medium access control protocols for ad hoc networks,” *IEEE Communications Surveys Tutorials*, vol. 6, pp. 2–16, First 2004.
- [5] L. XingGuo, W. JunFeng, and B. LinLin, “LEACH Protocol and Its Improved Algorithm in Wireless Sensor Network,” in *2016 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*, pp. 418–422, Oct 2016.
- [6] P. McDermott-Wells, “What is Bluetooth?,” *IEEE Potentials*, vol. 23, pp. 33–35, Dec 2005.
- [7] A. Chandra, V. Gummalla, and J. O. Limb, “Wireless Medium Access Control Protocols,” *IEEE Commun. Surveys*.
- [8] N. Abramson, “The ALOHA System,” *AFIPS 1970 Fall Joint Computer Conference*, vol. 37, pp. 281 – 285, 1970.

- [9] L. G. Roberts, “ALOHA packet system with and without slots and capture,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 5, no. 2, pp. 28 – 42, 1975.
- [10] H. Okada, Y. Igarashi, and Y. Nakanishi, “Analysis and application of framed ALOHA channel in satellite packet switching networks-FADRA method,” *Electron. Commun. Jpn.*, vol. 60, pp. 72 – 80, 1977.
- [11] W. Crowther, R. Rettberg, D. Walden, S. Ornstein, F. Heart, B. Beranek, and N. Inc, “A system for broadcast communication: Reservation-ALOHA,” *Proceedings of the International Conference on Systems Sciences*, pp. 371 – 374, 1973.
- [12] G. N. M. Sudhakar, N. D. Georganas, and M. Kavehrad, “Slotted Aloha and Reservation Aloha Protocols for Very High-speed Optical Fiber Local Area Networks Using Passive Star Topology,” *IEEE/OSA Journal of Lightwave Technologies*, vol. 9, pp. 1411 – 1422, 1991.
- [13] F. Borgonovo, A. Capone, M. Cesana, and L. Fratta, “RR-ALOHA a Reliable R-ALOHA broadcast channel for ad-hoc inter-vehicle communication networks,” *Proceedings of Med-Hoc-Net 2002*, 2002.
- [14] H. A. Cozzetti and R. Scopigno, “RR-Aloha+ A Slotted and Distributed MAC Protocol for Vehicular Communications,” *In IEEE Vehicular Networks Conference*, 2009.
- [15] J. Romkey, “Toast of the IoT: the 1990 interop internet toaster,” *IEEE Consumer Electronics Magazine*, vol. 6, no. 1, pp. 116 – 119, 2017.
- [16] K. Ashton, “That Internet of Things thing,” *RFiD Journal*, 2009.
- [17] H. Sundmaeker, P. Guillemin, P. Friess, and S. Woelffl, “Vision and challenges for realizing the Internet of Things,” *Cluster of European Research Projects on the Internet of Things CERP IoT*, 2010.
- [18] D. Evans, “The internet of things: How the next evolution of the internet is changing everything,” *CISCO white paper*, vol. 1, no. 1, pp. 1–11, 2011.

- [19] M. Q. ud din, A. Hazmi, L. F. D. Carpio, A. Goekceoglu, B. Badihi, P. Amin, A. Larmo, and M. Valkama, "Duty Cycle Challenges of IEEE 802.11ah Networks in M2M and IoT Applications," *European Wireless 2016; 22th European Wireless Conference*, pp. 1–7, 2016.
- [20] A. Dunkels, "The ContikiMAC Radio Duty Cycling Protocol," *SICS Tech. Rep.*, 2011.
- [21] J. Saraswat and P. P. Bhattacharya, "Effect of duty cycle on energy consumption in wireless sensor networks," *International Journal of Computer Networks and Communications*, vol. 5, no. 1, p. 125, 2013.
- [22] R. C. Carrano, D. Passos, L. C. Magalhaes, and C. V. Albuquerque, "Survey and taxonomy of duty cycling mechanisms in wireless sensor networks," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, pp. 181–194, 2014.
- [23] J. Li, W. Zeng, and A. Arora, "Achievable Throughput in Duty-Cycled Wireless Networks," *2012 IEEE 9th International Conference on Mobile Ad-Hoc and Sensor Systems (MASS 2012)*, pp. 290–298, 2012.
- [24] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 3, pp. 1567–1576 vol.3, 2002.
- [25] T. van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems*, 2003.
- [26] A. Rajandekar and B. Sikdar, "A Survey of MAC Layer Issues and Protocols for Machine-to-Machine Communications," *IEEE INTERNET OF THINGS JOURNAL*, vol. 2, no. 2, pp. 175 – 186, 2015.
- [27] E. Lutz, "Slotted ALOHA multiple access and error control coding for land mobile satellite networks," *International Journal of Satellite Communications and Networking*, vol. 5, pp. 275 – 281, 1992.

- [28] W. Zeng, S. Vasudevan, X. Chen, B. Wang, A. Russell, and W. Wei, "Neighbor discovery in wireless networks with multipacket reception," *Proc. 12th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2011.
- [29] F. V. Gallego, J. Alonso-Zarate, and L. Alonso, "Energy and delay analysis of contention resolution mechanisms for machine-to-machine networks based on low-power WiFi," *Proc. IEEE Int. Conf. Commun.*, pp. 2235 – 2240, 2013.
- [30] X. Liu, "Efficient unknown tag identification protocols in large-scale RFID systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, pp. 3145 – 3155, Dec 2014.
- [31] Y. Zhu, W. Jiang, Q. Zhang, and H. Guan, "Energy-efficient identification in large-scale RFID systems with Handheld reader," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, pp. 1211 – 1222, May 2014.
- [32] "Radio-frequency identity protocols class-1 generation-2 UHF RFID protocol for communications at 860 MHz-960 MHz version 1.0.9 EPCglobal Inc.," vol. 12, 2005.
- [33] S. s. Lam, "Packet broadcast networksA performance analysis of the R-ALOHA protocol," *IEEE Transactions Computing*, vol. 29, pp. 596 – 603, 1980.
- [34] M. K. Hossaina, S. Datta, S. I. Hossaina, and J. Edmonds, "ResVMAC: A Novel Medium Access Control Protocol for Vehicular Ad hoc Networks," *Procedia Computer Science*, vol. 109, pp. 432 – 439, 2017.
- [35] H. A. Cozzetti, R. M. Scopigno, L. Casone, and G. Barba, "Comparative Analysis of IEEE 802.11p and MS-Aloha in Vanet Scenarios," *2009 IEEE Asia-Pacific Services Computing Conference*, pp. 64 – 69, 2009.
- [36] D. J. Goodman and S. X. Wei, "Efficiency of packet reservation multiple access," *IEEE Transactions for Vehicular Technologies*, vol. 40, pp. 170 – 176, 1991.

- [37] F. Vazquez-Gallego, J. Alonso-Zarate, A. M. Mandalari, O. Briante, A. Molinaro, and G. Ruggeri, “Performance Evaluation of Reservation Frame Slotted-ALOHA for Data Collection M2M Networks,” *European Wireless 2014 20th European Wireless Conference*, pp. 1 – 6, 2014.
- [38] F. Vazquez-Gallego, L. Alonso, and J. Alonso-Zarate, “Modeling and Analysis of Reservation Frame Slotted-ALOHA in Wireless Machine-to-Machine Area Networks for Data Collection,” *Sensors (Basel, Switzerland)*, vol. 15, pp. 3911 – 3931, 2015.
- [39] S.-R. Lee, S.-D. Joo, and C.-W. Lee, “An Enhanced Dynamic Framed Slotted ALOHA Algorithm for RFID Tag Identification,” *In Proceedings of the Second Annual International Conference on Mobile and Ubiquitous Systems, Networking and Services*, pp. 166 – 172, 2005.
- [40] J.-C. Zhang and J.-J. Chen, “Design and Simulation of Reserved Frame-Slotted Aloha Anti-collision Algorithm in Internet of Things,” *In: Hu W. (eds) Electronics and Signal Processing. Lecture Notes in Electrical Engineering*, vol. 97, pp. 355 – 362, 2011.