Delay Sensitive Services In Multichannel Random Access Networks With Selfish Users

QIU CHAN LUO

Master of Science Thesis
Stockholm, Sweden 2012

TRITA-ICT-EX-2012:214
DELAY SENSITIVE SERVICES IN
MULTICHANNEL RANDOM ACCESS NETWORKS
WITH SELFISH USERS

Master Thesis Report

Student: Qiuchan Luo
Supervisor: Ali Özyagci
Examiner: Jens Zander
Abstract

Because of the scarcity of spectrum, it is becoming more and more important to use the spectrum efficiently. Dynamic spectrum allocation schemes are proposed to increase the efficiency of spectrum usage. Meanwhile, delay-sensitive services like voice service are still a strong contributor to operator revenues and constitute a significant portion of user demand. Therefore, it is valuable to study whether delay-sensitive services like voice service can be provided in multichannel random access network, in which multiple operators use the unlicensed spectrum and decentralized dynamic spectrum allocation scheme is adopted.

In this study, two User Reinforcement Learning schemes are proposed to model the behavior of selfish users in the multichannel random access network. We analyze the performance of these two schemes and compare them with Operator Reinforcement Learning scheme and Operator Non-Reinforcement Learning scheme in terms of delay, jitter, packet loss rate, consecutive packet loss rate, throughput per channel and utility per user.

We find that both the “User Reinforcement Learning Non-Retransmission” scheme and the “Operator Reinforcement Learning” scheme can provide a satisfactory quality service under a moderately high load.
Acknowledgments

I owe my deepest gratitude to my supervisor, Ali Özyagci, for his valuable suggestions and patient guidance during the planning and development of my thesis work. I would also like to express my great appreciation to Professor Jens Zander for providing so many valuable comments on my thesis work. My great thanks are also extended to Shen Sun for being my opponent during the thesis presentation.

I would like to offer my special thanks to my family, especially my parents, for their endless support and encouragement throughout my higher education. I would also like to thank my uncle and aunty for inspiring me to further my education in Sweden.

I would also like to extend my thanks to all my friends in Sweden. The life in Sweden would be boring without your companion and I hope our friendship lasting forever.

Finally, I would like to thank my boyfriend, Jeffrey Shen Quan, for always being supportive of me and bearing my mood swing.
Contents

Abstract .............................................................................................................................................. 2
Acknowledgments .......................................................................................................................... 3
Contents ............................................................................................................................................ 4
1. Introduction ................................................................................................................................... 6
  1.1 Dynamic Spectrum Access ....................................................................................................... 6
  1.2 Quality of Service Requirements of Delay Sensitive Services .............................................. 6
  1.3 Previous Work ....................................................................................................................... 7
  1.4 Problem definition ................................................................................................................ 8
  1.5 Thesis Outline ....................................................................................................................... 10
2. System Model ............................................................................................................................. 11
3. Mechanism Design ..................................................................................................................... 14
  3.1 Reinforcement Learning ......................................................................................................... 14
  3.2 User-RL Schemes ................................................................................................................ 15
    3.2.1 Utility Model of User-RL scheme .................................................................................... 15
    3.2.2 Learning the Probabilities ............................................................................................... 16
    3.2.3 Deciding Transmission Strategy .................................................................................... 17
  3.3 Operator-RL Scheme ............................................................................................................ 18
    3.3.1 Utility Model of Operator-RL scheme ........................................................................... 18
    3.3.2 Learning the Probabilities ............................................................................................. 19
    3.3.3 Scheduling the Users .................................................................................................... 19
4 Performance Metrics ..................................................................................................................... 22
  4.1 Average Packet Loss Rate .................................................................................................... 22
  4.2 Consecutive Packet Loss ...................................................................................................... 22
  4.3 Average Delay (latency) ....................................................................................................... 23
  4.4 Average Jitter ....................................................................................................................... 23
  4.5 Throughput per Channel per Time Slot ............................................................................... 24
  4.6 Utility per User per Time Slot ............................................................................................. 24
5. Methodology ............................................................................................................................... 25
  5.1 Learning Process .................................................................................................................. 25
5.2 Simulation Parameters ................................................................. 26
6. Simulation Results ........................................................................... 28
6.1 Average Packet Loss Rate ............................................................... 28
6.2 Consecutive Packet Loss ................................................................. 30
6.3 Average Delay (Latency) ................................................................. 31
6.4 Delay Distribution ........................................................................... 33
6.5 Average Jitter .................................................................................. 34
6.6 Throughput per Channel and Utility per User ..................................... 36
  6.6.1 Throughput per Channel per Time Slot ........................................ 37
  6.6.2 Utility per User per Time Slot ................................................... 38
7. Conclusion ......................................................................................... 41
8. Future Work ....................................................................................... 43
9. References ......................................................................................... 44
Appendix ............................................................................................... 45
1. Introduction

1.1 Dynamic Spectrum Access

With the increasing demand for accessing mobile broadband, it is very important to manage the limited radio spectrum resources efficiently. Fixed spectrum allocation policies have been adopted by most agencies which makes some parts of the radio spectrum rather crowded while some other parts are used infrequently. Because available spectrum is limited and fixed spectrum allocation policies make the spectrum usage sometimes inefficient, there is a need to find alternative mechanisms for managing spectrum in a more efficient way. Dynamic spectrum access (DSA) has been proposed to solve this problem [1]. Dynamic spectrum access (DSA) tries to allocate the spectrum to be more frequently used according to the variation of spectrum demand in time and space [2].

DSA schemes can be classified according to a number of criteria. With respect to the cooperation among the users in the DSA system, the DSA scheme can be either cooperative or competitive. In a cooperative system, users choose their transmission strategies to reach a common goal like maximizing the system sum utility. In a competitive system, users are considered to be selfish and they only choose their transmission strategies to maximize their own performance. This competitive system is also referred to as selfish system. The DSA scheme can also be classified as centralized or decentralized. If the DSA scheme is centralized, there is a central control entity which allocates spectrum resources to the users. Centralized system seems the most efficient by squeezing most number of bits for a given amount of bandwidth, but it requires lots of information exchanges among users, which increases the system complexity. Furthermore, sending controlling information to the network coordinator causes more delays for delay-sensitive services like voice services than decentralized system, which does not require any information exchange. In decentralized systems, there is no central control entity and users make the decision on when to access the spectrum resources by their own. Therefore, decentralized DSA scheme may be more preferable than centralized one in terms of reducing system complexity and can be applied to future wireless network.

1.2 Quality of Service Requirements of Delay Sensitive Services

Delay sensitive services like voice services are still a strong contributor to operator revenues and constitute a significant portion of user demand, so it is valuable to study whether delay sensitive services like voice services can be provided in such a future wireless network, in which multiple operators use the unlicensed spectrum and decentralized DSA scheme is adopted. If certain quality of service requirement is satisfied, then we can say that the delay sensitive services, for example voice services, can be provided in this decentralized multi-channel random access network. The quality of service requirement of a delay-sensitive service is represented as a set of constraints. There are different delay-sensitive services, such as voice service, video service and so on. For different delay-sensitive services, the set of constraints can be different. To satisfy the quality of service requirement of voice service, four things
needed to be considered: packet loss rate, consecutive packet loss, latency and jitter.

Packet loss happens when there is too much traffic and it will cause interruption in voice communication. Consecutive packet loss means the loss of multiple adjacent packets. The same amount of consecutive packet loss causes more damage to the call quality than that of the packet loss at random time instances. Packet Loss Concealment (PLC) is a technique used to mask the effects of lost or discarded packets. PLC is generally effective only for small numbers of consecutive lost packets and low packet loss rate. The acceptable packet loss rate is 20% [3] and the acceptable consecutive packet loss is a total of 20-30 milliseconds of speech [4]. In other words, if less than 20% of packets and 20-30 milliseconds of speech and are lost, users can still have reasonable voice communication quality by applying certain PLC concealment [5].

Latency means the delay of packet delivery, in other words, it means the time from a packet is generated to it is received. Large delay should be avoided because it will cause bad echoes and degrade the quality of communication. For delay-sensitive service like VoIP service, the delay is made up of coding delay, serialization delay, queuing delay, propagation delay, de-jitter delay and decoding delay. Coding delay depends on the type of codec used and for codec G.711, the coding delay is almost 0 whereas for G.729, the coding delay is 15 milliseconds [6]. Serialization delay is the time to place a packet on the transmission link and is decided by the speed of link and the size of a packet. For example, for a 218-byte packet, it takes 12.5 milliseconds time to place this packet on a 128 Kbps link. Queuing delay happens at switching point of network like routers and ranges from a few milliseconds to hundreds of milliseconds. Propagation delay depends on the distance between the packet sender and receiver. De-jitter delay is the jitter buffer to allow for variable packet arrival time and often ranges from 50 to 100 milliseconds. Decoding delay means the time to decode the signal and is often half the coding delay [6].

The variance of delay is defined as jitter. High levels of jitter cause large numbers of packets to be discarded by the jitter buffer in the receiving IP phone or gateway. This may result in severe degradation in call quality or large increases in delay.

In brief, if the packet loss rate, consecutive packet loss, latency and jitter requirement are satisfied, we can say delay-sensitive services like voice service can be provided in this system.

1.3 Previous Work

Distribution of packet delay in multichannel random access network has been studied previously. In [7], the author considers decentralized multichannel random access systems with users which have statistical channel state information. The results show that lack of information and selfish behavior of each user causes performance degradation compared to the performance of the users in a scheduling system or cooperative system. However, the performance is evaluated only in terms of throughput which is not applicable to delay sensitive services. In [8], the problem of autonomous users deploying delay-sensitive applications in decentralized
multi-channel random access networks has been studied. In this study, each user is assumed to know the total number of users in the system. A conjecture-based channel selection scheme has been proposed, which means the user can determine an optimal action based on the linear belief function to minimize its own delay over all possible wireless channels. The results show that the average delay of the system is around 10 milliseconds. However, this study does not show the packet loss rate and jitter of this decentralized system. Furthermore, it is not so realistic for each user to know the total number of users in the system in decentralized MRA network because in decentralized system the users are not able to know the existence of each other and there is not any central entity to inform them of this as well. Moreover, it might be more interesting to take energy consumption into consideration, which means the user in this system may choose to wait when the channel condition is very bad. By doing this, the total energy cost of a user will be reduced in the long run which is beneficial for users.

A multichannel random access network with incomplete information can be regarded as a multi-agent system without supervision [9]. In this system, an agent doesn’t have any information about the other agents’ actions but the reward of his action is affected by the other agents. Therefore, the agent interacts with the environment that includes everything outside the agent frequently and learns to optimize his action decision that can maximize his payoff and this process is called reinforcement learning. Previous study show that reinforcement learning based schemes can be used to solve resource allocation problems in multichannel random wireless networks. In [10], a best response learning algorithm has been proposed to improve the user’s bidding strategy in a cognitive radio network. The secondary users(SUs) simultaneously and selfishly bid for the resource based on their current states, experienced environment, and estimated future reward at each stage. The learning algorithm is similar to Multi-agent reinforcement Q-learning established in [11] except that they consider the impacts of other SUs’ bidding actions through the state classification and transition probability approximation. However, a central spectrum moderator is needed to allocate resources to the SUs, which is different from our scenario. In [12], a distributed discrete repeated power control game in CR networks is modeled as an N-user nonzero sum game. The scenario of this work is quite similar to ours; that is the network is multichannel random access network with selfish users. However, in their study, each secondary user adopts the Bush-Mosteller reinforcement algorithm established in [13] to maximize their own throughput by adjusting their transmission power while each user needs to maximize their payoff by deciding when to transmit a packet in our study.

1.4 Problem definition

The objective of the thesis is to design reinforcement learning based schemes that can be used to implement a decentralized multi-channel random access network to provide delay sensitive services in unlicensed spectrum, to evaluate the performance of these schemes under different traffic loads, and to answer whether the quality of
service requirement of such services can be satisfied or not under different traffic loads. More exactly, we design two “User Reinforcement Learning Schemes”: one is “User Reinforcement Learning Retransmission Scheme”, in which user can retransmit a lost packet within an allowed duration of time and the other is “User Reinforcement Learning Non-retransmission Scheme”, in which user can’t retransmit a lost packet. In both schemes, all users in the system are selfish and will learn the environment to modify their transmission strategies to maximize their payoff. To simplify the notation, we use “User-RL” to denote “User Reinforcement Learning”, “User-RL-Retrans” to denote the “User Reinforcement Learning Retransmission” and “User-RL-NonRetrans” to denote the “User Reinforcement Non-retransmission”.

We consider a selfish decentralized multi-channel random access network, in which users are assumed to only know their own exact channel condition. In other words, they have incomplete information of the system. As described previously, this system is a multi-agent system without supervision. Therefore, a multi-agent reinforcement-learning scheme is used to model the learning behavior of selfish users in the decentralized system. With respect to the quality of service requirement, the basic metrics for the performance of the decentralized system adopting the proposed scheme are average delay (latency), jitter, packet loss rate and consecutive packet loss rate. To have a more comprehensive view of the system, we study the delay distribution of packet delivery, throughput per channel, utility per user. Though the basic performance metrics considered here are most relevant to voice services, the performance results are not only valid for voice services but also valid for the other delay sensitive services.

In unlicensed spectrum, it is also possible that operators compete against each other to maximize their individual payoffs. More exactly, users served by the same operator in the system cooperate with each other and it is the operators that are doing the resource management. Then we imagine how multiple operators provide delay-sensitive services in the unlicensed spectrum as references of our proposed User-RL schemes. The worst case is that multiple operators schedule the users randomly in the unlicensed spectrum as they did in licensed spectrum; we call this as “Operator Non-Learning Scheme”. It can be predicted that there will be a lot of collisions because operators do not learn how the other operators schedule their users. A better case is that we regard it as a multi-agent system without supervision but the learning agent is the operator instead of the user. Each operator interacts with the environment constantly to adjust their scheduling of users to maximize their overall payoff. We call this as “Operator Reinforcement Learning Scheme”. To simplify the notation, we use “Operator-NL” to denote “Operator Non-Learning” and “Operator-RL” to denote “Operator Reinforcement Learning”.

In our study, we are going to compare the User-RL schemes with Operator-RL scheme and Operator-NL scheme in terms of the metrics mentioned above.

This study answers the following questions:

- Can the performance of the decentralized system adopting the
User-RL-Retrans scheme and User-RL-NonRetrans scheme satisfy the quality of service requirement of voice services? If so, what is the maximum traffic load that the two proposed schemes can support while satisfying the quality of service requirement of voice services and which scheme is better?

• Which performance is better, if we compare the User-RL schemes with Operator-NL scheme and Operator-RL scheme?

1.5 Thesis Outline

The report is structured as follows: Section 2 introduces system model used in this work; Section 3 gives a thorough description of the proposed mechanism; Section 4 defines the performance metrics that we used in the study to analyze the system performance; Section 5 describes the simulation method and parameters we used in the study; Section 6 includes the analysis of the results obtained; Section 7 contains the conclusion; Section 8 proposes what needs to be solved in the future; Section 9 lists the references of our work.
2. System Model

A wireless network scenario is considered in our study. In our scenario, the system is decentralized without a central control entity and DSA scheme is employed. We assume there are \( N \) users and \( K \) channels in the system. We assume that the location of the user in the system don’t change so much that his shadow fading doesn’t change too much. However, his Rayleigh fading changes. The \( N \) users can belong to \( M \) operators. If \( M \) equals 1, it means all the users in the network belong to the same operator. If \( M \) is larger than 1, it means the users in the network belong to multiple different operators. As shown in Figure 1, \( N \) equals 12, \( K \) equals 6 and \( M \) equals 2, which means there are 12 users, 6 channels and 2 operators in the network. The traffic load of the system shown in Figure 1 is \( N/K=12/6=2 \) users/channel. More specifically, a selfish slotted-ALOHA system with multiple channels is considered to reflect the random access nature of this DSA system. In the selfish slotted-ALOHA system, each user knows the total number of channels and his respective path loss of these channels, but he doesn’t know the path loss of the other users and the actions that the other users have taken or will take. Each user in the system competes for transmitting on the channel where he could successfully transmit a packet and get better payoff. Each user will always be inclined to transmit on the channel where he would use less energy cost and has the highest probability of successful transmission. In each time slot, every user chooses to transmit on one channel or wait to transmit in a later time slot. As shown in Figure 1, User 4, User 6 and User 8 choose to wait in this time slot while the other users all choose to transmit on a certain channel. In each time slot, if only one user tries to transmit on one channel, then the packet from this user can be transmitted successfully. If more than one user transmits on the same channel simultaneously, collision would happen and the packets would be lost. As shown in Figure 1, User 1, User 5, User 9 and User 12 all choose to transmit on channel 1 in this time slot, so collision happens and packets of these four users will get lost. Meanwhile, only User 7 chooses to transmit on channel 3, so the packet of this user can be successfully transmitted.
The length of a time slot is fixed and should be no less than the length of packet transmission time. Packet transmission time means the time that a packet is transmitted to the time that a packet is received, which is made up of the time to place the packet on the transmission link, the time that a packet needs to wait at the router or gateway and the propagation time of the packet. The time to place the packet on the transmission link is the time spent in the MAC layer and the time that a packet needs to wait at the router or gateway and the propagation time of the packet are the time spent in the network layer. In our model, we just consider the time to place the packet on the transmission link and the propagation time to simplify the analysis. Meanwhile, in a cellular network, the propagation time can be as small as several microseconds; therefore, it can be ignored in our model. As a result, we only consider the time to place the packet on the transmission link to represent the transmission time of a packet. As described previously, the time to place the packet on the transmission link depends on the packet size and the speed of the transmission link. Let $S$ denote the packet size, $R$ denote the speed of transmission link and $T$ denote transmission time, and then we can obtain:

$$T = \frac{S}{R}$$  \hspace{1cm} (2-1)

The packet size ($S$) represents the number of bytes (or bits) that are filled into a packet, which is determined by the device that is able to encode or decode a digital data. This device is called codec. There are many types of codec standards and different codec standard specifies different packet size. G.711 codec is commonly used and causes very small coding delay [6]. We simply take the packet size defined in G.711 codec in our system as a representative of packet sizes of other possible codec. In the G.711 codec, the packet size ($S$) is 218 bytes [14]. In our study, we assume the speed of transmission link ($R$) is 1 Mbps and then we can obtain the transmission time of a
packet as: 

\[ (218 \times 8/10^6) \times 10^3 = 1.744 \text{ milliseconds} \]

As described previously, the length of a time slot should be no less than the length of a packet transmission time, therefore, we assume the length of a time slot is 2 milliseconds. According to the G.711 codec, a packet is generated every 20 milliseconds. To simplify the analysis, we assume that a packet has to be transmitted before a new packet is generated, or it will be dropped. As a result, a packet can exist in the system for at most 20 milliseconds which equals 10 time slots and we denote this as the survival threshold of a packet \( (T_{thr}) \). We also regard 10 time slots as a frame. As shown in Figure 2, packet delay (latency) is measured from the moment that the packet is generated to the moment that the packet is received successfully. Packet 1 is transmitted successfully right after it is generated; therefore, the delay of Packet 1 equals 1 time slot. The delay of packet 2 is 4 time slots because this packet is transmitted three time slots after it is generated. This is because the channel condition in the first three time slots is very bad and the channel condition in the fourth time slot becomes good which makes the user wait for three time slots and transmit Packet 2 in the fourth time slot. Packet 3 is in the same case.

![Figure 2 codec and packet delay](image-url)
3. Mechanism Design

As mentioned in the Introduction part, we have designed two User-RL schemes. Meanwhile, we consider Operator-RL scheme and Operator-NL scheme as references of the proposed schemes. In User-RL schemes and Operator-RL scheme, we use reinforcement learning algorithm to model the user’s or the operator’s learning behavior. In Operator-NL scheme, the operator schedules its users’ transmission without learning.

3.1 Reinforcement Learning

In a multi-agent system without supervision, an agent doesn’t have any information about the other agents’ actions but the reward of his action is affected by the other agents. Therefore, the agent interacts with the environment that includes everything outside the agent frequently and the agent learns to optimize his action decision that can maximize his utility. This type of learning is called reinforcement learning as shown in Figure 3.

In a state of environment, the agent takes an action and receives reward and updates the state to a different one. Let $S$ denote the set of all possible states and $A$ denote the set of all possible actions. The initial state can be randomly selected from the set of all possible states $S$. Let $s_t \epsilon S$ denotes the state of environment at time $t$, $s_{t+1} \epsilon S$ denotes the state of environment at time $t+1$, $a^t \epsilon A$ denote the action that the agent takes at time $t$ and $r_t$ denote the received reward. When an agent takes action $a^t$ at state $s_t$, he receives a reward $r_t$ and moves to the next state $s_{t+1}$. The received reward is a random variable because it varies from state to state and we denote it as $R$. At time $t$, the agent learns to select an action from the set of all possible actions $A$ which can maximize the expected reward $E[R]$ at this state [9]. In our study, we model the state of environment as a set of successful transmission probabilities of all channels in the network. The successful transmission probability of a channel means how likely a user can transmit a packet successfully on this channel. For example, if the successful transmission probability of a user on a certain channel is 0.5, then we can say there is a probability of 0.5 for this user to transmit a packet successfully on this channel. As described above, the agent is responsible for updating the state of environment; therefore in our case, the agent is responsible for updating the successful transmission probabilities.

![Figure 3 general reinforcement learning process](image-url)
In both User-RL schemes, the user is the learning agent; whereas in the Operator-RL scheme, the operator is the learning agent.

3.2 User-RL Schemes
In a multi-channel random access network with selfish users, the user is regarded as the learning agent and each user learns to update the successful transmission probabilities independently after each transmission. We design two User-RL schemes: one is User-RL-NonRetrans scheme and the other is User-RL-Retrans scheme. In User-RL-NonRetrans scheme, if collision happens, then the packet will be lost and will not be retransmitted in the same frame; in User-RL-Retrans scheme, if collision happens, the packet will be retransmitted in the same frame until it has been successfully transmitted or a new packet is generated. Figure 6 shows the transmission behavior of a certain user in the system employing the two schemes respectively. In Figure 4, Packet 1 and Packet 2 denote two different packets containing different information.

![Diagram showing transmission behavior](image)

**Figure 4 User-RL-NonRetrans and User-RL-Retrans Scheme**

3.2.1 Utility Model of User-RL scheme

For successful transmission, the user gets a positive payoff which is normalized as 1, while the user receives a zero payoff for unsuccessful transmission or no transmission. Let \( e_n^k \) denote the transmission cost, which represents the \( n^{th} \) user’s transmission cost on \( k^{th} \) channel. Once the user decides to transmit, he has to spend a transmission cost even though the packet is transmitted unsuccessfully, which has a negative effect on the utility.

In User-RL scheme, the user makes transmission decision every time slot. As said previously, a packet is assumed to have a survival threshold \( T_{thr} \). As time approaches the survival threshold \( T_{thr} \), say in the \( 10^{th} \) time slot, practically the selfish user will prefer to transmit the packet instead of waiting, because after the survival threshold, the packet will be dropped directly. So in User-RL scheme, the delay factor affects the user’s transmission decisions. We model the delay effect as a discounting factor \( D_s \) on the transmission cost when a user fails to transmit a packet. This means the transmission cost for a failed transmission will be reduced as time approaches \( T_{thr} \), which will encourage the user to transmit instead of waiting as time approaching \( T_{thr} \).
Delay Sensitive Services in Multichannel Random Access Networks with Selfish Users

\( D_s \) is modeled as below:

\[
D_s = \gamma^{(s-1)}
\]  

(3-1)

Where \( 0 \leq \gamma < 1 \). The factor \( \gamma \) is a constant which is determined by a specific application [15] and in our system, we choose it as 0.65. \( s \) is the number of time slots a packet has existed in the system before it expires, and \( 1 \leq s \leq 10 \).

Taken transmission cost and delay effect into consideration, we can define the user’s utility as follows:

\[
\begin{align*}
  u_n^k &= \begin{cases} 
    1 - e_n^k & \text{transmit and succeed} \\
    D_s \times (-e_n^k) & \text{transmit and failed} \\
    0 & \text{wait}
  \end{cases} \\
  &\quad \text{if the } n^{th} \text{ user successfully sends out a packet and the packet has existed for } m \text{ time slots in the system, then the user will receive a non-discounted payoff. However, if the } n^{th} \text{ user unsuccessfully sends out a packet and the packet has existed } s \text{ time slots in the system, then the user will receive a discounted negative payoff. Based on this utility model, at each time slot, the user will be inclined to transmit on the channel where he could spend less energy and has highest probability of successful transmission.}
\end{align*}
\]  

(3-2)

The transmission cost \( e_n^k \) is derived as a ratio of \( P_t \) and \( P_{max} \) [2]. In the definition of the transmission cost, \( P_t \) is the transmit power that just satisfies the SNR requirement for a successful transmission, whereas \( P_{max} \) is the maximum transmit power, which just satisfies the SNR requirement for a successful transmission at the cell border. The transmission cost is simplified as below:

\[
e_n^k = \frac{(r/r_0)^\alpha (f_k/f_{max})^\beta}{S \times R}
\]  

(3-3)

In the equation, \( r \) is the distance from the user to the base station; \( r_0 \) is the cell radius and we choose it as 200 m; \( \alpha \) is the path loss exponent and we choose it as 3 to model the urban area; \( f_k \) is the frequency of \( k^{th} \) channel; \( f_{max} \) is the maximum frequency among all channels; \( \beta \) is the dependence of attenuation on frequency and normally it is chosen as 2; \( S \) is the shadow fading component which is log-normally distributed with unit mean and standard deviation \( \sigma \), which is selected as 4 dB; \( R \) is the fast fading component which is exponentially distributed with unit mean.

### 3.2.2 Learning the Probabilities

The way of learning the successful transmission probabilities is same for both schemes and is described below. Let \( P^t_n = \{ p_{n1}^t, p_{n2}^t, \ldots p_{nK}^t \} \) denote the transmission probabilities set of the \( n^{th} \) user is at time slot \( t \) and \( K \) denote the total number of channels in the system. \( p_{nk}^t \) means the successful transmission probability of the \( n^{th} \)
user is on channel $k$ at time slot $t$. In other words, the $n^{th}$ user has a probability of $p_{nk}^t$ to transmit a packet successfully on channel $k$ at time slot $t$. The initial value of $P_{nk}^t$ is chosen randomly by the $n^{th}$ user itself. Based on Bush-Mosteller Reinforcement learning algorithm [13], the successful transmission probabilities of the $n^{th}$ user is updated according to the following formula:

$$
p_{nk}^{t+1} = \begin{cases} 
p_{nk}^t + l_{nk} \cdot s_{nk}^t \cdot (1 - p_{nk}^t), & \text{if } s_{nk}^t \geq 0 \\
p_{nk}^t + l_{nk} \cdot s_{nk}^t \cdot p_{nk}^t, & \text{if } s_{nk}^t < 0 
\end{cases}
$$

(3-4)

Where $l_{nk}$ is the learning rate and $0 < l_{nk} < 1$. $s_{nk}^t$ is the stimulus factor that is defined as follows:

$$
s_{nk}^t = \begin{cases} 
\frac{N_{sk}^t}{N_{tk}^t} - p_{nk}^t, & \text{transmit on channel } n \\
0, & \text{otherwise}
\end{cases}
$$

(3-5)

$N_{sk}^t$ means the total number of successfully transmitted packets on channel $k$ of the $n^{th}$ user until time slot $t$. $N_{tk}^t$ means the total number of transmitted packets on channel $k$ of the $n^{th}$ user until time slot $t$. The larger is the change between the probabilities in this slot and the updated probabilities in the next time slot, the higher is $s_{nk}^t$. The higher is the learning rate $l_{nk}$, the larger is the change between the probabilities in this slot and the next slot.

### 3.2.3 Deciding Transmission Strategy

The way of deciding transmission strategy for both schemes is also same and is described below. At the time slot $t$, expected utility of the $n^{th}$ user on channel $k$ is defined below:

$$
E[U_{nk}^t] = p_{nk}^t \cdot (1 - e_{nk}^t) + (1 - p_{nk}^t) \cdot \left(-D_s \cdot e_{nk}^t\right)
$$

(3-6)

Where $e_{nk}^t$ means the energy cost of the $n^{th}$ user on channel $k$ at the time slot $t$, and it can be calculated according to equation 2-4. $D_s$ is the delay discounting factor defined in equation 2-2.

We assume that the user can only know his own energy costs on all channels at a certain time slot and he doesn’t know how many users there are in the system and the actions of other users. Meanwhile, we assume that in the first frame, when to generate a packet in the first frame is randomly determined by the users and in the later frames, packets are generated every 20 milliseconds. The purpose of this assumption is to avoid unrealistic periodic collisions. Taken voice service as an example, it is more realistic that each user starts to make phone calls in the system at a random instant than that all of them start to make phone calls at the same time.
The $n^{th}$ user uses the following condition to decide whether to transmit at time slot $t$ or not:

$$k^* = \arg\max_k \mathbb{E}[U_{nk}^t]$$

$$\mathbb{E}[U_{nk*}^t] > 0$$

If the above condition is satisfied, the user will decide to transmit the packet on channel $k^*$. If the above condition is not satisfied, the operator will decide to wait and transmit in later time slot. If a user waits constantly for more than 10 time slots, the packet will be dropped.

### 3.3 Operator-RL Scheme

In Operator-RL scheme, the operator learns to compete against the other operators by scheduling its users’ transmission to maximize its overall expected utility in a frame. The utility definition in Operator-RL scheme is slightly different from that of User-RL scheme.

#### 3.3.1 Utility Model of Operator-RL scheme

For successful transmission, the user gets a positive payoff which is normalized as 1, while the user receives a zero payoff for unsuccessful transmission or no transmission. In Operator-RL scheme, we also use $e_{nk}^k$ to denote the transmission cost, which represents the $n^{th}$ user’s transmission cost on $k^{th}$ channel. The definition of $e_{nk}^k$ is same as equation 3-3 in User-RL scheme.

We assume that the operator can know the energy costs of all his users in a frame; in other words, the operator will have some knowledge of the channel conditions in the future. This is not practical, but this will give us an upper bound on the performance. Based on this assumption, in each frame, the operator schedules the user to transmit on a certain channel at a certain time slot of this frame where the user has smallest transmission cost and highest probability of successful transmission. Therefore, we can define the user’s utility in the Operator-RL as follows:

$$u_{nk}^k = \begin{cases} 
1 - e_{nk}^k & \text{transmit and succeed} \\
-e_{nk}^k & \text{transmit and failed} \\
0 & \text{wait}
\end{cases}$$

As described in Section 2, we regard 10 time slots as a frame. Let $K$ denote the total number of channels in the system. Then there are $K \times 10$ slots in each frame. Based on this utility model, in one frame, the operator can schedule the user to transmit on one of the $K \times 10$ slots where the user can spend less energy cost and has highest probability of successful transmission.
3.3.2 Learning the Probabilities

Let \( p_m^f = \{p_{m1}^f, p_{m2}^f, ..., p_{mK}^f\} \) denote the transmission probabilities set of the \( m^{th} \) operator in frame \( f \) and \( K \) denote the total number of channels in the system. \( p_{mk}^f \) means the successful transmission probability of the \( m^{th} \) operator on channel \( k \) at frame \( f \). In other words, the users of the \( m^{th} \) operator have a probability of \( p_{mk}^f \) to transmit a packet successfully on channel \( k \) in frame \( f \). The initial value of \( p_m^f \) is chosen randomly by the \( m^{th} \) operator itself. Based on Bush-Mosteller Reinforcement learning algorithm [13], the successful transmission probabilities of the \( m_{th} \) operator is updated according to the following formula:

\[
p_{mk}^{f+1} = \begin{cases} 
  p_{mk}^f + l_{mk} \cdot s_{mk}^f \cdot \left(1 - p_{mk}^f\right), & \text{if } s_{mk}^f \geq 0 \\
  p_{mk}^f + l_{mk} \cdot s_{mk}^f \cdot p_{mk}^f, & \text{if } s_{mk}^f < 0 
\end{cases} \tag{3-9}
\]

Where \( l_{mk} \) is the learning rate and \( 0 < l_{mk} < 1 \). \( s_{mk}^f \) is the stimulus factor that is defined as follows:

\[
s_{mk}^f = \begin{cases} 
  \frac{NS_{mk}^f}{Nt_{mk}^f} - p_{mk}^f, & \text{transmit on channel } k \\
  0, & \text{otherwise} 
\end{cases} \tag{3-10}
\]

\( NS_{mk}^f \) means the total number of successfully transmitted packets on channel \( k \) of the \( m^{th} \) operator until the \( f^{th} \) frame. \( Nt_{mk}^f \) means the total number of transmitted packets on channel \( k \) of the \( m^{th} \) operator until the \( f^{th} \) frame. The larger is the change between the probabilities in this slot and the updated probabilities in the next time slot, the higher is \( s_{mk}^f \). The higher is the learning rate \( l_{mk} \), the larger is the change between the probabilities in this slot and the next slot.

Based on the above learning scheme, each operator keeps updating the successful transmission probabilities until they become reliable and stable. When the successful transmission probabilities become stable, then operators can make more accurate decision based on those probabilities. We assume that each operator updates the successful transmission probabilities every frame.

3.3.3 Scheduling the Users

We assume that \( N_m \) represents the number of users served by the \( m^{th} \) operator.
If there are \( K \) channels in the system, then the operator can assign any one of the \( K \times 10 \) slots to a user to avoid collision among these \( N_m \), which is shown in Figure 5; however, we can see there are still collisions among users from different operators from Figure 5.

![Figure 5 Operator Reinforcement Learning Scheme](image)

In order to maximize the overall utility, the operator will try to assign the slot to the user in which he can have the largest expected utility. At the \( i^{th} \) time slot of the \( f^{th} \) frame, expected utility of the \( n^{th} \) user of the \( m^{th} \) operator on channel \( k \) is defined below:

\[
E[U_{mk}^n] = p_{mk}^f \cdot (1 - e_{mk}^n) + (1 - p_{mk}^f) \cdot (-e_{mk}^n) \quad (3-11)
\]

Where \( e_{mk}^n \) means the energy cost of the \( n^{th} \) user of the \( m^{th} \) operator on channel \( k \) at the \( i^{th} \) time slot of the \( f^{th} \) frame and it can be calculated according to equation 2-4. \( i \) ranges from 1 to 10 since there are 10 time slots in a frame.

In the mean time, the users are synchronized by the operator so that each user generates a packet in the beginning of a frame, which makes it easier for the operator the schedule the users. Then the operator can assign the slot to the \( n_{th} \) user that satisfies the following condition:

\[
\begin{align*}
(i^*, k^*) &= \arg \max_{i, k} E[U_{mk}^n] \\
E[U_{mk}^{n^*}] &> 0
\end{align*}
\quad (3-12)
\]

If the above condition is satisfied, then the packet of the \( n^{th} \) user of the \( m^{th} \) operator will be transmitted on channel \( k^* \) at time slot \( i^* \) of the \( f^{th} \) frame. If the above condition is not satisfied, the expected utility of the user will be zero or negative no matter which action it is going to take. If the expected utility is zero, the operator can either choose to schedule the user’s transmission or simply drop the packet of this user. In our study, we assume that the operator will decide to drop the
packet of the $n^{th}$ user to avoid using transmission cost. If the expected utility is less than zero, the operator will decide to drop the packet of the $n^{th}$ user to avoid negative utility being added to the overall utility and in the mean time, transmission cost is also avoided. Therefore, the operator will decide to drop the packet of the $n^{th}$ user if the above condition is not satisfied.

To simplify the simulation, we also assume that the operator uses greedy algorithm to schedule the users in each frame. Let $N_{rs}$ denote the available number of slots that haven’t be assigned to a user by the $m^{th}$ operator in the $f^{th}$ frame. Let $N_u$ denote the number of users that haven’t be scheduled by the $m^{th}$ operator in the $f^{th}$ frame. The initial value of $N_{rs}$ is $K \times 10$. In our study, $N_{rs}$ is assumed to be larger than $N_u$. The greedy algorithm works in the following procedure:

- **Step 1**: the $m^{th}$ operator randomly selects a user from the $N_u$ users and calculates the expected utility of this user over $N_{rs}$ available slots.

- **Step 2**: Decide the user’s transmission schedule based on 3-4 and update :
  - If the condition 3-4 is satisfied, schedule the user to transmit. $N_{rs} = N_{rs} - 1$, $N_u = N_u - 1$
  - If the condition is not satisfied, the packet of the will be discarded. $N_u = N_u - 1$

- **Step 3**: Repeat Step 1 and Step 2 until all the users have been scheduled.

The greedy algorithm will not always give the transmission schedule that will maximize the sum utility; however the sum utility obtained using this algorithm will be more and more close to the maximum sum utility as the number of users in the system increases.
4 Performance Metrics

As said in the problem definition part, there are several performance metrics used in the study to evaluate the performance of the selfish MRA system that employs proposed schemes.

4.1 Average Packet Loss Rate

A packet will be considered lost in two conditions. One is when collision happens, the packet is lost. The other condition is that a packet without transmission will be dropped after the survival threshold $T_{thr}$ by the user and will also be regarded as a lost packet. Assuming in certain time duration, the total number of generated packets of the $n^{th}$ user is denoted as $G_n$ and the total number of lost packets of the $n^{th}$ user is denoted as $L_n$. $N$ is the total number of users in the system. Then the average packet loss rate ($Plrate_\text{aver}$) can be calculated as:

$$Plrate_\text{aver} = \frac{\sum_{n=0}^{N} L_n}{N} \times 100\% \quad (4-1)$$

4.2 Consecutive Packet Loss

As shown in Figure 6, if there are three adjacent lost packets that contain different voice information in a certain period of time, then we say the number of consecutive lost packets in this period is 3. If there is only one lost packet in a certain period of time, then we say the number of consecutive lost packets during this period is 0.

Assuming in certain time duration, the total number of consecutive lost packets of the $n^{th}$ user is denoted as $CL_n$. Then added up all the consecutive lost packets of the $n^{th}$ user, $CL_n$ is obtained. As described in the System Model part, we simply take the packet size defined in G.711 codec in our system as a representative of packet sizes of other possible codec. In the G.711 codec, the packet size ($S$) is 218 bytes which equals 20 milliseconds of speech [14]. Then the average time duration of consecutive packet loss per user ($CL$) is calculated as below:

$$CL = \frac{\sum_{n=1}^{N} CL_n}{N} \times 20 \text{ (milliseconds)} \quad (4-2)$$
4.3 Average Delay (latency)

As described in the Introduction part, packet delay consists of coding delay, serialization delay, propagation delay, network queuing delay, de-jitter buffer delay and decoding delay. In our model, we only consider delay in the MAC layer and that means we will ignore network queuing delay and de-jitter buffer delay. Meanwhile, we do not consider coding delay and decoding delay in the system. Since the propagation time of a packet in cellular network is also very small, propagation delay can be ignored as well. In our model, delay in the MAC layer is made up of the period of time that a packet waits in the MAC layer after it is generated and the packet transmission time. Furthermore, we only consider the delay of the successfully transmitted packet to calculate the average delay because delay means the delay of packet delivery, in other words, it means the time from a packet is generated to it is successfully transmitted. The delay of the successfully transmitted $m^{th}$ packet of the $n^{th}$ user is defined as $D_{n}^{m}$. Assuming in certain time duration, the total number of packets successfully transmitted by the $n^{th}$ user is denoted as $M_{n}$. $N$ is the total number of users in the system. Then the average delay of the system ($Delay_{aver}$) can be obtained by the following equation:

$$ Delay_{aver} = \frac{\sum_{n=1}^{N} \sum_{m=1}^{M} D_{n}^{m}}{N} \text{ (milliseconds)} $$

(4-3)

4.4 Average Jitter

Jitter means the variation in latency, or packet delay. Consistent or similar packet delay is good for reliable voice services [6]. In our study, we calculate the Jitter by taking the absolute value of difference between adjacent packet delay samples and summing those differences up and then dividing by the number of delay samples (minus 1). This jitter measurement is called absolute jitter [16].

Let’s take Figure 2 as an example, there are three delay samples: delay1, delay2 and delay3. Delay1 equals 1 time slot. Delay2 equals 4 time slots. Delay3 equals 5 time slots. The length of a time slot is 2 milliseconds as defined previously. Then the jitter is calculated as:

$$ \frac{|delay2 - delay1| + |delay3 - delay2|}{3 - 1} \cdot 2 = \frac{|4 - 1| + |5 - 4|}{2} \cdot 2 = 4 ms $$

Using the method described above, we can calculate the jitter of each user. Assuming in certain time duration, the Jitter of the $n^{th}$ user is defined as $J_{n}$ and then the average jitter ($J_{aver}$) is obtained as:

$$ J_{aver} = \frac{\sum_{n=1}^{N} J_{n}}{N} \text{ (milliseconds)} $$

(4-4)
4.5 Throughput per Channel per Time Slot

Throughput per channel indicates the efficiency of the system from a resource utilization point of view. It is defined as the number of successfully transmitted packets per time slot. Let $I$ denote the total number of time slots. The total number of packets successfully transmitted in the $I$ time slots by the $n^{th}$ user is denoted as $N_{s}^{n}$. There are $K$ channels and $N$ users in the system. The throughput is defined as the number of successfully transmitted packets per time slot. Therefore, the throughput per channel ($Thr_{averc\_h}$) is defined as:

$$Thr_{averc\_h} = \frac{\sum_{n=1}^{N} N_{s}^{n}}{K \cdot I}$$  \hspace{1cm} (4-5)$$

4.6 Utility per User per Time Slot

As mentioned previously, throughput results indicate the efficiency of the system from a resource utilization point of view. In contrast, the utility per user indicates the efficiency of the system as it is perceived by the users. The utility of the $n^{th}$ user in the $i^{th}$ time slot is denoted as $u_{n}^{i}$, which is calculated by formula 2-3. Let $I$ denote the total number of time slots. Then the utility per channel per time slot ($U_{aver}$) is calculated as:

$$U_{aver} = \frac{\sum_{n=1}^{N} \sum_{i=1}^{I} u_{n}^{i}}{N \cdot I}$$  \hspace{1cm} (4-6)$$
5. Methodology

We implement a Mote-Carlo simulator. All users are randomly located in the area in every realization. In one realization, we assume that the location of the user in the system does not change so much that his shadow fading does not change too much. However, his Rayleigh fading changes in every time slot. We also assume that the algorithm will converge faster than the change in the shadow fading. We compare the performance of the schemes under different scenarios. Different scenario has different number of channels and different number of users.

As described in the previous section, in each time slot, the users are making the decisions that will maximize their expected utilities when the successful transmission probabilities become stable. Meanwhile, in each frame, the operators are making the decisions that will maximize their overall expected utilities when the successful transmission probabilities become stable. Let’s assume at the time slot $t$, the successful transmission probabilities become stable; then the results obtained after the time slot $t$ can be used to reflect the system performance. Therefore, in the beginning, we need to study the learning process and try to find out this time point.

5.1 Learning Process

We take a look at the two-user-two-channel case as an example to illustrate how the learning algorithm progresses in time. The result is shown in Figure 7. We obtain the learning process of the two users by using Operator-RL scheme, User-RL-NonRetrans scheme and User-RL-Retran scheme respectively. As shown in Figure 8, for the three schemes, the successful transmission probabilities changes a lot in the first 400 time slots, changes slowly from $400^{th}$ time slot to $1000^{th}$ time slot. After $1000^{th}$ time slots, there still exists minor (approximately $\pm 0.2$) difference between each time slot because the players are still updating their probabilities based on the received reward. Because the difference is very small, the successful probabilities can be considered to be stable after $1000^{th}$ time slot; then results obtained after the $1000^{th}$ time slot can be used to reflect the system performance. We observe similar results for the other simulation scenarios. Therefore, we will focus on studying the results obtained after the $1000^{th}$ time slot.

In the User-RL-NonRetrans scheme, the successful transmission probabilities of both users on both channels are very high. There are two reasons. One is that in User-RL-NonRetrans scheme, there are fewer collisions. Another reason is that there are just two users while there are two channels (20 time slots for each user to transmit a packet in a frame), which also reduces the collision.

In the User-RL-Retran scheme and Operator-RL scheme, the user 1 (operator 1) will choose to transmit on channel 1 and user 2 (operator 2) will choose to transmit on channel 2 for most of the time after $1000^{th}$ time slot, so collision is reduced which means both players will receive positive reward for most of the time slots. Therefore, the transmission probability will become more and more stable.
5.2 Simulation Parameters

The simulation parameters are shown in Table 1.

Table 1: Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value/Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell Radius((r_0))</td>
<td>200 m</td>
</tr>
<tr>
<td>Nr. Of Channels((K))</td>
<td>2,4,8,16</td>
</tr>
<tr>
<td>Traffic load((T))</td>
<td>1,2,4,8,16</td>
</tr>
<tr>
<td>Nr. Of Users((N))</td>
<td>(K \times T)</td>
</tr>
<tr>
<td>Nr. Of Operators((M))</td>
<td>2</td>
</tr>
<tr>
<td>Standard Deviation of Shadow Fading</td>
<td>4 dB (\text{dB})</td>
</tr>
<tr>
<td>Mean Value of Rayleigh Fading</td>
<td>1</td>
</tr>
<tr>
<td>Minimum Frequency of Channels((f_{\text{min}}))</td>
<td>500 MHz</td>
</tr>
<tr>
<td>Maximum Frequency of Channels((f_{\text{max}}))</td>
<td>(f_{\text{min}} \times K)</td>
</tr>
<tr>
<td>Path Loss Exponent((\alpha))</td>
<td>3</td>
</tr>
<tr>
<td>Dependence of Attenuation on Frequency((\beta))</td>
<td>2</td>
</tr>
<tr>
<td>Delay Discounting factor((D_s))</td>
<td>(0.65^{(s-1)}), (1 \leq s \leq 10, s \text{ is an integer})</td>
</tr>
<tr>
<td>Learning Rate((\eta_{ik}))</td>
<td>0.65</td>
</tr>
</tbody>
</table>
Under a certain combination of $K$ and $T$, there will be 400 simulation results. Then by averaging this 400 simulation results, we obtain the final results of this combination.

In each realization, the successful transmission probabilities will be updated 1200 times and we take the results in the last 200 time slots and average them to obtain the final result of this realization.
6. Simulation Results

In this section, the performance of the User-RL-NonRetrans scheme, User-RL-Retrans scheme, Operator-RL scheme and Operator-NL scheme are compared in terms of average delay, average jitter, packet loss rate and consecutive packet loss. The throughput per channel and utility per user are also analyzed in this section to study the efficiency of the system from a resource utilization point of view and from the perspective of user. Noteworthily, the results are not only valid for voice services but also valid for a more general class of delay-sensitive services, although the performance metrics considered in the study are most relevant to voice services.

In this section, we give results for the simulation cases in which there are 8 channels in the system. The complete set of results for 2, 4 and 8 channels are in the appendix.

6.1 Average Packet Loss Rate

Ideally, there shouldn’t be any packet loss for good quality of voice services. If there are some lost packets and the number of the lost packets is within a certain range and the total length of consecutive packet loss is acceptable, then a technique called Packet Loss Concealment (PLC) can be used to mask the effects of lost packets. Practically, if no more than 20% of packets are lost, users can still have reasonable voice communication quality by applying PLC[3]. Therefore, we study the average packet loss rate of the system over different traffic load and we select the 8-channel scenario as an example to show the results of the packet loss rate. The result of the 8-channel scenario is shown in Figure 8. The confidence interval is shown in Table 1.

As shown in Figure 8, when the traffic load is no more than 8 users per channel, Operator-RL scheme and User-RL-NonRetrans scheme can give a satisfactory packet loss rate which is less than 20%. Operator-NL Scheme can only give satisfactory packet loss rate when the traffic load is less than 4 users per channel because if two operators randomly schedule the users, there is high probability that they schedule the users to transmit in the same time slots and then more collisions happen and more packets are lost compared to the other schemes. For User-RL-Retrans scheme, it gives similar performance as User-RL-NonRetrans scheme for very low traffic loads (≤ 2 users/channel) and then the User-RL-NonRetrans scheme always gives a packet loss rate larger than 20%.

In User-RL-Retrans scheme, every user can transmit a packet several times in a frame which increases the number of collisions compared to User-RL-NonRetrans scheme. For example: there are three users. User 1 and User 2 decide to transmit a packet for the first time in the 1st time slot in the same frame while User 3 decides to transmit a packet for the first time in the 10th time slot in the same frame. In the User-RL-NonRetrans scheme, the packets of User 1 and User 2 will be lost because of collision while the packet of User 3 can be successfully transmitted. However, in User-RL-Retrans scheme, User 1 and User 2 may decide to transmit their packets for a second time in the 10th time slot and the packet of User 3 will be lost for good.
It can also be noticed that packet loss rate increases as traffic load increases. There are two reasons accounting for this. One is that as traffic load increases, more and more collisions happen. Another reason is as traffic load increases, the successful transmission probabilities will be reduced so more users will choose not to transmit their packets to save energy. Those un-transmitted packets will be dropped when new packets are generated and they are part of the lost packets. When the traffic load is 16, the system will have very high packet loss rate. Even if the system applying Operator-RL scheme, it will have a packet loss rate nearly 70%.

From Figure 8, we can conclude that the Operator-RL scheme and User-RL-NonRetrans scheme can be applied to provide satisfactory quality of service with respect to packet loss rate under a moderately low traffic load. Operator-NL scheme and User-RL-Retrans scheme are not appropriate to be employed since they can just have acceptable packet loss rate when the traffic load is very small (<=2).

Figure 8  Packet Loss Rate Comparison between User-RL, Operator-RL and Operator-NL
### 6.2 Consecutive Packet Loss

If we want to employ the Packet Loss Concealment (PLC) technique, the acceptable consecutive packet loss should be no more than 30 milliseconds of speech as described in [4]. Therefore, we also study the consecutive packet loss of the four schemes. As we did previously, we select the 8-channel scenario to show the consecutive packet loss of the system. The result is shown in Figure 9 and the confidence interval is shown in Table 2.

As shown in Figure 9, the length of the consecutive packet loss per user is no more than 30 milliseconds for all the schemes when the traffic load is no more than 8 users per channel. When the traffic load is 2 users per channel, the consecutive packet loss for the four schemes are quite similar. When the traffic load is 16 users per channel, the Operator-RL scheme and User-RL-NonRetrans scheme result in consecutive packet loss less than 30 milliseconds. Because the confidence interval of the consecutive packet loss of the Operator-NL scheme shows the consecutive packet loss can be as high as 30.08 milliseconds, we can’t say it gives acceptable consecutive packet loss though the figure shows it gives a consecutive packet loss around 30 milliseconds. Normally, if a scheme gives less packet loss rate, it also gives less consecutive packet loss. Operator-RL scheme has the smallest packet loss rate as shown in Figure 8. This also explains why the Operator-RL scheme has smallest consecutive packet loss when the traffic load is larger than 2 users per channel.

From Figure 9, we can also see that the User-RL-Retrans scheme has larger consecutive packet loss than that of User-RL-NonRetrans scheme. When traffic load is larger than 8, the User-RL-Retrans scheme will give a consecutive packet loss larger than 30 milliseconds. This is because retransmission leads to more collisions. And we can also explain this with respect to Packet Loss Rate. Because the packet loss rate of User-RL-Retrans scheme is much higher than that of User-RL-NonRetrans scheme, there is large probability that the consecutive packet loss of User-RL-Retrans scheme is larger than that of User-RL-NonRetrans scheme.

As shown in Figure 9, as traffic load increases, the consecutive packet loss increases. One reason is that more and more collisions happen as traffic load increases and more packets are lost which may increase the consecutive packet loss. Another reason is more and more packets will be dropped without transmission by the users.
because the successful transmission probabilities become smaller as the traffic load increases.

From Figure 9, we can conclude that Operator-RL scheme and User-RL-NonRetrans scheme are more acceptable with respect to consecutive packet loss criteria that we used.

![Figure 9](image_url)

**Figure 9** Consecutive Packet Loss Rate Comparisons among User-RL-NonRetrans scheme, User-RL-Retrans scheme, Operator-RL scheme and Operator-NL scheme

<table>
<thead>
<tr>
<th>Traffic-load Schemes</th>
<th>1 users/channel</th>
<th>2 users/channel</th>
<th>4 users/channel</th>
<th>8 users/channel</th>
<th>16 users/channel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operator-RL</td>
<td>[0.1943, 0.1957]</td>
<td>[0.7070, 0.7155]</td>
<td>[2.6536, 2.6726]</td>
<td>[9.4710, 9.5192]</td>
<td>[29.9870, 30.0898]</td>
</tr>
<tr>
<td>Operator-NL</td>
<td>[0.0033, 0.0033]</td>
<td>[0.0067, 0.0067]</td>
<td>[0.0162, 0.0163]</td>
<td>[0.0668, 0.0669]</td>
<td>[23.4057, 23.4095]</td>
</tr>
<tr>
<td>User-RL-Retrans</td>
<td>[0.0034, 0.0034]</td>
<td>[0.0652, 0.0657]</td>
<td>[14.8972, 14.9570]</td>
<td>[34.7355, 34.7461]</td>
<td>[39.1561, 39.1565]</td>
</tr>
<tr>
<td>User-RL-NonRetrans</td>
<td>[0.0033, 0.0034]</td>
<td>[0.0221, 0.0229]</td>
<td>[0.8683, 0.8755]</td>
<td>[7.1833, 7.2121]</td>
<td>[25.9200, 25.9225]</td>
</tr>
</tbody>
</table>

**6.3 Average Delay (Latency)**

For voice services, the acceptable delay should be no more than 150 ms according to ITU_T G.114 standard [6]. As described in section 4.1, we only consider delay in the MAC layer delay and ignore the network queuing delay and de-jitter buffer delay. As described previously, the MAC layer delay considered in our model is made up of waiting time in the MAC layer and transmission time of a packet. In reality, the delay
is largely caused by network queuing delay and de-jitter buffer delay which can be added up to 100 ms or more [6]. Therefore, the MAC layer delay considered in our model will be far less than the 150ms acceptable delay proposed by ITU-T G.114 standard. The queuing delay in the network is less controllable than the MAC layer delay because it depends largely on the how busy the traffic is; therefore, to keep the MAC layer delay as small as possible is necessary to provide good quality service. If the network queuing delay and de-jitter buffer delay add up to 90 ms, then the MAC layer delay can be 60 ms or less. If the network queuing delay and de-jitter buffer delay add up to 130 ms, then the MAC layer delay should be kept less than 20 ms to provide good quality service. In our case we are going to study the average MAC layer delay over different traffic loads of these four schemes to study which scheme provide smallest packet delay.

The results are shown in Figure 10 and we only select the 8-channel scenario to analyze the effect of the proposed schemes on average delay of the system. The confidence interval is shown in Table 3. As shown in the figure, all the schemes can provide a delay less than 14 milliseconds and User-RL-NonRetrans scheme leads to the smallest latency. It can also be observed in Figure 10 that Operator-RL scheme and Operator-NL scheme have much higher delay than User-RL-NonRetrans scheme. In the Operator-RL scheme and Operator-NL scheme, operator schedules the user to transmit, so the user can be schedule to transmit a packet 8 time slots after it is generated. By doing this, the delay of this packet is increased. In the long run, the overall delay increases thus the average delay increases as well. For the User-RL-Retrans scheme, it leads to very small delay when the traffic load is no more than 2 users per channel.

From Table 3, we can also see that the average delay of the two User-RL schemes increases as traffic load increases. When there are more users in the system, if all users transmit at the same time, a lot of collisions will happen. Therefore, some users learn to transmit several time slots after a packet is generated, which reduces the collisions while increasing the average delay. The increase is more obvious for User-RL-Retrans scheme, because in User-RL-Retrans scheme, the increase in delay is not only caused by the increase of number of users but also by retransmission. If collision happens and the packet is lost in 1st time slot, the packet can still be retransmitted in a later time slot, say the 8th time slot. In this case, the delay of this packet is $8 \times 2 = 16$ milliseconds; therefore, the delay increases because of retransmission.
To have a better understanding of the average delay results, we can take a look at the delay distribution at the 32-user-8-channel case, which means the traffic load is 4 users per channel. The delay distribution of user 1 of operator 1 is shown in Figure 11.

As shown in Figure 11, the packet delay of User 1 is one time slot for most of the packets in the User-RL-NonRetrans scheme, which explains why User-RL-NonRetrans scheme has the smallest average delay in Figure 10. Note that in the system that employs User-RL schemes, each user generates a packet at different starting point as we described in section 3.2, which means they don’t need to generate a packet in the first time slot of each frame, unlike the users in the Operator-RL scheme and Operator-NL scheme, which generate packets in at beginning of each frame. Therefore, the delay of most packets is 1 time slot doesn’t
mean that they are transmitted in the first time slot of a frame; instead it means most packets is transmitted successfully one time slot after it is generated. In the Operator-RL and Operator-NL scheme, the packet delay can be as large as 10 time slots; this is because in these two schemes, operator schedules the user to transmit based on his knowledge of User 1’s energy cost in a frame. So the user can be scheduled to transmit in the 1st time slot in one frame while the same user may be scheduled to transmit in the 8th time slot in another frame.

Figure 11 Delay distribution of user 1 of operator 1 at the 32-user-8-channel case

6.5 Average Jitter

Jitter is an important indicator in measuring the quality of voice services, so we are going to study the average jitter of the system by applying the four schemes. The acceptable level of jitter is determined by the de-jitter buffer size of the system. If the de-jitter buffer size is 50 milliseconds, then the jitter should be no more than 50 milliseconds; otherwise, the packet will be discarded. Normally, the de-jitter buffer size is around 50-100 milliseconds [3] so the acceptable jitter should be less than 100 milliseconds in most cases. As we said previously, network queuing delay can ranges from several milliseconds to hundreds of milliseconds which is the main factor that causes jitter. Since we just consider the MAC layer delay in our analysis, the jitter we obtained will be far less than 100 milliseconds. Even though the variance of MAC layer delay is not the main reason for the jitter, it is still necessary to keep it as small as possible. Therefore, we compare the jitter of the system by applying the four schemes and same as we did in the analysis of delay effect, we only select 8-channel scenario as an example and the result is shown in Figure 12 and the 95% confidence
As shown in Figure 12 and Table 4, all of the four schemes can result in a jitter less than 8 milliseconds. As we said previously, we didn’t consider the jitter caused by the network queuing delay; therefore, the jitter of the system by applying the four schemes are all far less than 100 milliseconds. But we can still observe that the User-RL schemes lead to a much smaller jitter than the Operator-RL scheme and Operator NL scheme. The systems employing Operator-RL scheme and Operator-NL scheme have very high jitter, which are around 6.5 milliseconds. Because in the system applying Operator-RL scheme or Operator-NL scheme, a user can have a delay of 2 milliseconds in one frame while have a delay of 10 milliseconds in another frame when operator schedules the user to transmit; then the delay of a user can vary from frame to frame. The time slot allocation of the operator doesn’t try to minimize the jitter; instead, the operators allocate the users to the time slots to maximize the overall expected utility. So the variance of delay, or jitter, can be very high. If the operator tries to minimize the jitter, then the users would be scheduled to transmit at a time slot that is not so advantageous in terms of transmission power.

For User-RL-Retrans scheme, the jitter increases obviously as the traffic load increases but it is still less than that of the system employing Operator-RL scheme. The User-RL-NonRetrans scheme has smallest jitter. In User-RL-NonRetrans scheme, the delay of a user is very small in the first place and doesn’t change a lot so the jitter of the system employing this scheme is very small. We can use the results obtained in Figure 11 to explain this as well. As shown in Figure 11, in User-RL-NonRetrans scheme, most of the packets are transmitted 1 time slot after they are generated, therefore, the variance of the delay, or jitter, is small. We can also see that the jitter of both schemes increases as traffic load increases. When there are more users in the system, the successful transmission probabilities will decrease and the user can only transmit a packet in the time slots in which he has very good channel condition. For example, he may transmit a packet one time slot after it is generated in one frame and 8 time slots after it is generated in another frame. That’s why the variance of delay increases as traffic load increases.
Figure 12 Jitter comparisons among User-RL-NonRetrans scheme, User-RL-Retrans scheme, Operator-RL scheme and Operator-NL scheme

Table 4: 95% Confidence interval of the jitter

<table>
<thead>
<tr>
<th>Traffic-load Schemes</th>
<th>1 users/channel</th>
<th>2 users/channel</th>
<th>4 users/channel</th>
<th>8 users/channel</th>
<th>16 users/channel</th>
</tr>
</thead>
<tbody>
<tr>
<td>User-RL-Retrans</td>
<td>[0.0017,0.0017]</td>
<td>[0.0094,0.0094]</td>
<td>[0.4111,0.4172]</td>
<td>[1.2252,1.2322]</td>
<td>[1.9482,1.9880]</td>
</tr>
<tr>
<td>User-RL-NonRetrans</td>
<td>[0.0001,0.0001]</td>
<td>[0.0001,0.0001]</td>
<td>[0.0001,0.0001]</td>
<td>[0.0001,0.0001]</td>
<td>[0.0037,0.0037]</td>
</tr>
</tbody>
</table>

6.6 Throughput per Channel and Utility per User

As mentioned previously, throughput results indicate the efficiency of the system from a resource utilization point of view. The throughput per channel is studied first in this section and the utility per user indicate the efficiency of the system as it is perceived by the users. Therefore, we are going to study the throughput per channel and utility per user in the following sections in order to analyze the system performance more comprehensively.
6.6.1 Throughput per Channel per Time Slot

Throughput is defined as the number of successfully transmitted packets per time slot. Throughput per channel is defined as the number of successfully transmitted packets per time slot per channel. The larger is the throughput per channel; the better is the system performance. Figure 13 shows the throughput per channel of the system when there are 8 channels and the confidence interval is shown in Table 5.

As shown in Figure 13, the four schemes can give similar throughput per channel under low traffic load (<=4 users per channel). As shown in Table 4, the throughput per channel of Operator-RL scheme is slightly higher than the other three schemes when the traffic load is 4. Under high traffic load (>=8 users per channel), the Operator-RL scheme gives a noticeably higher throughput per channel. In Operator-RL scheme, there is no collision among users belonging to one operator so more packets can be successfully transmitted which increases the throughput.

Though the throughput per channel of User-RL-NonRetrans scheme is not as high as that of Operator-RL scheme, it also gives a relatively high throughput per channel. The throughput per channel of User-RL-NonRetrans scheme is always higher than that of Operator-NL scheme because in the Operator-NL scheme, two operators schedule the users’ transmission randomly which leads to lots of collisions.

We can also observe from Table 5, the throughput per channel of User-RL-Retran scheme is slightly better than that of User-RL-NonRetrans scheme when the traffic load is no more than 2 users per channel. This is because when the traffic load is low, there are not so many users in the system, which gives chance for the user to retransmit a packet successfully. However, as traffic load increases to 4 users per channel, the advantage of User-RL-NonRetrans scheme become obvious over User-RL-Retran scheme. This is because as the number of users increases in the system, the number of collisions increases. Retransmission will add additional collisions to the system which will degrade the system performance.

Another interesting finding is the throughput per channel increases in the beginning and then decreases as the traffic load increases which is different from the trend of packet loss rate. Two reasons account for this. One is as traffic load increases, the number of collisions increases. Another reason is more and more packets will be dropped without transmission when the traffic load is higher than 8 users per channel, so the number of transmitted packets at this traffic load is smaller than that when the traffic load is 8 users per channel. As a result, the number of successfully transmitted packets is smaller when the traffic load is 16 users per channel than that when the traffic load is 8 users per channel. This explains why the throughput per channel decreases as the traffic load increases from 8 users per channel to 16 users per channel.

From Figure 13, we can also conclude that the Operator-RL scheme and the User-RL-NonRetrans scheme are more appropriate to be applied in open spectrum
because they provide high throughput per channel.

![Graph showing throughput comparison between User-RL, Operator-RL and Operator-NL schemes](image)

**Figure 13** Throughput per Channel Comparisons among User-RL-NonRetrans scheme, User-RL-Retrans scheme, Operator-RL scheme and Operator-NL scheme

**Table 5:** 95% Confidence interval of Throughput per Channel

<table>
<thead>
<tr>
<th>Traffic-load Schemes</th>
<th>1 users/channel</th>
<th>2 users/channel</th>
<th>4 users/channel</th>
<th>8 users/channel</th>
<th>16 users/channel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operator-RL</td>
<td>[0.0995, 0.0995]</td>
<td>[0.1803, 0.1804]</td>
<td>[0.3223, 0.3224]</td>
<td>[0.4845, 0.4851]</td>
<td>[0.3220, 0.3221]</td>
</tr>
<tr>
<td>Operator-NL</td>
<td>[0.0997, 0.0997]</td>
<td>[0.1995, 0.1995]</td>
<td>[0.398, 0.398]</td>
<td>[0.8090, 0.8091]</td>
<td>[0.6281, 0.6281]</td>
</tr>
<tr>
<td>User-RL-Retrans</td>
<td>[0.0997, 0.0997]</td>
<td>[0.1993, 0.1993]</td>
<td>[0.2205, 0.2216]</td>
<td>[0.0901, 0.0903]</td>
<td>[0.0296, 0.0296]</td>
</tr>
<tr>
<td>User-RL-NonRetrans</td>
<td>[0.0993, 0.0993]</td>
<td>[0.1962, 0.1962]</td>
<td>[0.3755, 0.3756]</td>
<td>[0.5875, 0.5879]</td>
<td>[0.4617, 0.4618]</td>
</tr>
</tbody>
</table>

**6.6.2 Utility per User per Time Slot**

As we said previously, in the User-RL-NonRetrans scheme and User-RL-Retrans scheme, user learns to find the best transmission strategy that can maximize the expected utility each time slot. In the Operator-RL Scheme, operator learns to find the best scheduling method that can maximize the sum expected utilities in every frame. As shown in section 3.2.1 and 3.3.1, we take energy cost into consideration when defining the utility model of both User-RL scheme and Operator-RL scheme. Based on these utility models, in User-RL scheme, the user learns to transmit on the channel...
where he could spend less energy cost at each time slot; in Operator-RL scheme, the operator schedules the user to transmit on the channel at certain time slot of each frame where the user could spend less energy cost. Therefore, the utility results capture the energy efficiency to some extent. From the perspective of the user, the utility per user per time slot can give them the information that how much utility he can receive by transmitting a packet in a frame. Therefore, we are going to study the utility per user in this section. Figure 14 shows the utility per user of the system when there are 8 channels and the confidence interval is shown in Table 6.

As shown in Figure 14, Operator-RL scheme always gives highest utility per user per time slot when the traffic load and the number of channels are the same. The User-RL-NonRetrans scheme gives higher utility per user than that of Operator-NL scheme. User-RL-NonRetrans scheme always give a higher utility per user per time slot than that of User-RL-Retrans scheme when the traffic load is larger than 4 users per channel. And we can also observe that the User-RL-Retrans scheme even gives a negative utility under high traffic load (>=8 users per channel). This means if a user decides to transmit in a system employing Retransmission User Learning Scheme under high traffic load (>=8), he is more likely to transmit unsuccessfully.

We can also observe that as traffic load increases, the utility per user per time slot decreases because it becomes more and more difficult for a user to transmit a packet successfully.

Figure 14 Utility per User Comparisons among User-RL, Operator-RL and Operator-NL
Table 6: 95% Confidence interval of Utility per User

<table>
<thead>
<tr>
<th>Traffic-load Schemes</th>
<th>1 users/channel</th>
<th>2 users/channel</th>
<th>4 users/channel</th>
<th>8 users/channel</th>
<th>16 users/channel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operator-RL</td>
<td>[0.0942,0.0946]</td>
<td>[0.0896,0.0897]</td>
<td>[0.07967,0.07968]</td>
<td>[0.0582,0.0583]</td>
<td>[0.0160,0.0161]</td>
</tr>
<tr>
<td>Operator-NL</td>
<td>[0.0946,0.0947]</td>
<td>[0.0957,0.0957]</td>
<td>[0.0957,0.0957]</td>
<td>[0.0933,0.0933]</td>
<td>[0.03071,0.03074]</td>
</tr>
<tr>
<td>User-RL-Retrans</td>
<td>[0.0931,0.0931]</td>
<td>[0.0936,0.0936]</td>
<td>[0.0446,0.0446]</td>
<td>[0.0043,0.00423]</td>
<td>[-0.0012,-0.0012]</td>
</tr>
<tr>
<td>User-RL-NonRetrans</td>
<td>[0.0942,0.0942]</td>
<td>[0.0936,0.0936]</td>
<td>[0.08607,0.08608]</td>
<td>[0.06439,0.0644]</td>
<td>[0.0230,0.02300]</td>
</tr>
</tbody>
</table>
7. Conclusion

In this thesis, we have studied how the delay sensitive services can be provided in the unlicensed spectrum and how the user can learn to select its transmission strategy in a selfish decentralized multichannel random access network. To solve this problem, we designed two User-RL Schemes: one is “User Reinforcement Learning Retransmission (User-RL-Retrans)” scheme and the other is “User Reinforcement Learning Non-retransmission (User-RL-NonRetrans)” scheme.

By studying the latency, jitter, packet loss rate and consecutive packet loss of the systems which employ the two User-RL schemes respectively, we find that the system that employ the User-RL-NonRetrans scheme can satisfy the quality of service requirement of delay-sensitive service like voice services when the traffic load is no more than 8 users per channel when there are 8 channels. More specifically, when the traffic load is no more than 8 users per channel, the User-RL-NonRetrans scheme can have an average delay less than 50 milliseconds, a jitter less than 2 milliseconds, a packet loss rate less than 20% and a total length of consecutive packet loss less than 30 milliseconds. However, the system employing User-RL-Retrans scheme can only satisfy the quality of service requirement when the traffic load is less or equal 2 users per channel. When the traffic load is larger than 2, the system employing User-RL-Retrans scheme will have a packet loss rate higher than 40%, which is not acceptable according to quality of service requirement. Therefore, the User-RL-NonRetrans scheme is more suitable to be employed in the selfish decentralized system.

By comparing the User-RL schemes with Operator-RL scheme and Operator-NL scheme, we find the one applying Operator-NL scheme has the worst performance. Though the Operator-RL scheme has better performance than User-RL-NonRetrans scheme in terms of packet loss rate, consecutive packet loss, throughput and utility, it gives larger latency and jitter than User-RL-NonRetrans scheme since scheduling in the Operator-RL scheme will increase the variance of delay. Both Operator-RL scheme and User-RL-NonRetrans scheme provide satisfactory quality of service, but User-RL-NonRetrans scheme would be more preferable to the Operator-RL scheme because it does not require the control signalling that is used in the Operator-RL scheme to schedule the users.

Because of the scarcity of the spectrum, it would be possible that operators decide to use unlicensed spectrum to provide services in the real world. If the operators provide delay-sensitive services in the unlicensed spectrum using the User-RL-NonRetrans scheme proposed in our study, the equipment cost will be reduced because the User-RL-NonRetrans scheme can provide satisfactory quality of service without requiring any controlling devices.

For the regulators, our study can be interesting. As the spectrum resources are limited, the regulator may try to release more spectrum for the unlicensed use; however, in unlicensed spectrum, the quality of service will be degraded. Therefore,
the regulator will need to study how the quality of service in unlicensed spectrum will be like in order to decide whether to open more spectrum or not. Our study can be a reference for the regulator to see how much the quality of delay-sensitive service will be degraded in the unlicensed spectrum.
8. Future Work

In our study, we made the delay discounting factor and the learning rate two as two constants and didn’t study the effect of these two factors on the system. It could be interesting to study how the system performance would be when these two factors change in the future. It could also be interesting to study how the interference outside the system affects the system performance and whether the system that employing the User-RL-NonRetrans scheme can still have acceptable quality of service if there are some users outside the system tries to access the channels in the system. Although the utility per user per time slot captures the energy efficiency to some extent, it is not so straightforward. Therefore, it could be useful to describe the energy efficiency in a more detailed way in the future.

Meanwhile, we evaluate the system performance in terms of latency, jitter, packet loss rate and consecutive packet loss respectively. Another way to evaluate the system performance is using mean opinion score. It could also be interesting to evaluate the system performance in terms of mean opinion score by using existing mean opinion score prediction model.
9. References


Appendix

In the appendix, we present the simulation results for all scenarios (there are 2, 4 or 8 channels in the system) with respect to packet loss rate, consecutive packet loss, average delay, jitter, throughput per channel and utility per user.

Figure A1 Packet Loss Rate Comparisons among User-RL-NonRetrans, User-RL-Retrans, Operator-RL and Operator-NL when there are 2, 4 or 8 channels

Figure A2 Consecutive Packet Loss Comparisons among User-RL-NonRetrans, User-RL-Retrans, Operator-RL and Operator-NL when there are 2, 4 or 8 channels

Figure A3 Average Delay Comparisons among User-RL-NonRetrans, User-RL-Retrans, Operator-RL and Operator-NL when there are 2, 4 or 8 channels
Figure A4 Average Jitter Comparisons among User-RL-NonRetrans, User-RL-Retrans, Operator-RL and Operator-NL when there are 2, 4 or 8 channels

Figure A5 Throughput per Channel Comparisons among User-RL-NonRetrans, User-RL-Retrans, Operator-RL and Operator-NL when there are 2, 4 or 8 channels

Figure A6 Utility per User Comparisons among User-RL-NonRetrans, User-RL-Retrans, Operator-RL and Operator-NL when there are 2, 4 or 8 channels