Modeling and Optimization of Adaptive Video Streaming over LTE-Unlicensed

Μοντελοποίηση και Βελτιστοποίηση Μετάδοσης Βίντεο μέσω LTE και Χρήση μη Αδειοδοτημένου Φάσματος

Master thesis
by
Galanopoulos Apostolos

University of Thessaly
Department of Electrical and Computer Engineering

Supervisors:

Argyriou Antonios
Korakis Athanasios
Potamianos Gerasimos

Volos, June 2016

Acknowledgments

I would like to thank my supervisor on this thesis Dr. Argyriou Antonios as well as George Iosifidis for their precious advice, inspiration and devotion towards helping me complete my thesis.

Dedicated to my family and friends.

Abstract

This work aims to tackle the problem of adaptive video streaming over a LTE network that utilizes the recently developed framework of Licensed Assisted Access, where users of the LTE network are opportunistically assigned with radio resources from unlicensed as well as licensed carriers through Carrier Aggregation. The unpredictable nature of the wireless channel as well as the unknown utilization of the unlicensed carrier by other unlicensed users constitute a challenging problem of selecting the highest possible video segment quality for each user while also trying to deliver the segments in time for playback, and thus avoiding buffer under-run events that deteriorate viewing experience. These two aspects of the problem are analyzed and algorithms are proposed to optimally select video quality in the first place, and secondly, to perform resource allocation in order to deliver the segments of the selected qualities in time. Moreover, a comparison is made with the typical proportional fair scheduler, as well as a state of the art adaptive video streaming framework, in terms of average segment quality and number of buffer under-run events in order to validate the effectiveness of the proposed algorithms under various unlicensed carrier traffic conditions. Results show that the proposed quality selection and scheduling algorithms, not only achieve higher video segment quality in most cases, but also minimize the amount and duration of video freezes as a result of buffer under-run events.

Περίληψη

Με την παρούσα εργασία επιχειρείται η αντιμετώπιση του προβλήματος της προσαρμοζόμενης μετάδοσης βίντεο μέσω ενός LTE δικτύου που χρησιμοποιεί το πρόσφατο Licensed Assisted Access στους χρήστες του οποίου εκχωρούνται ευκαιριακά πόροι από μη αδειοδοτημένο φάσμα, καθώς και από αδειοδοτημένο, μέσω της τεχνολογίας της συνάθροισης φερόντων. Η απρόβλεπτη φύση του ασύρματου καναλιού, καθώς και η άγνωστη χρήση του μη αδειοδοτημένου φέροντος από άλλους χρήστες διαμορφώνουν ένα δύσκολο πρόβλημα επιλογής της καλύτερης δυνατής ποιότητας των τμημάτων βίντεο για κάθε χρήστη, καθώς επίσης και της έγκαιρης μεταφοράς τους για αναπαραγωγή, αποφεύγοντας έτσι συμβάντα εκκένωσης του buffer τα οποία θα αλλοιώσουν την εμπειρία θέασης των χρηστών. Οι δυο προαναφερόμενοι παράγοντες του προβλήματος αναλύονται, και προτείνονται αλγόριθμοι αρχικά για τη βέλτιστη επιλογή της ποιότητας των τμημάτων του βίντεο, και έπειτα για την ανάθεση πόρων με στόχο την έγκαιρη μεταφορά των τμημάτων στην επιλεχθήσα ποιότητα. Τέλος γίνεται σύγκριση με την δημοφιλή τεχνική της αναλογικά δίκαιης εκχώρησης πόρων, καθώς και με μια σύγχρονη λύση για προσαρμοζόμενη μετάδοση βίντεο, ως προς τη μέση ποιότητα βίντεο και το πλήθος των συμβάντων εκκένωσης του buffer, έτσι ώστε να επιβεβαιωθεί η αποτελεσματικότητα των προτινόμενων αλγορίθμων σε διάφορα σενάρια χρήσης του μη αδειοδοτημένου φέροντος. Τα αποτελέσματα δείχνουν ότι οι προτεινόμενοι αλγόριθμοι επιλογής της ποιότητας και εκχώρησης πόρων, δεν επιτυγχάνουν μόνο καλύτερη ποιότητα των τμημάτων του βίντεο, αλλά επιπλέον ελαχιστοποιούν το πλήθος και τη διάρκεια των παγωμάτων στη ροή του βίντεο ως αποτέλεσμα των συμβάντων εκκένωσης του buffer.

Contents

List of Figures

List of Tables

List of Abbreviations

3GPP 3rd Generation Partnership Project

ADMM Alternating Direction Method of Multipliers

BCASP Backlog and Channel Aware Scheduling Policy

BER Bit Error Rate

CA Carrier Aggregation

CC Component Carrier

CCA Clear Channel Assessment

CDF Cumulative Distribution Function

CRC Cyclic Redundancy Check

CSI Channel State Information

DASH Dynamic Adaptive Streaming over HTTP

DCF Distributed Coordination Function

FDD Frequency Division Multiplexing

FSPL Free Space Path Loss

IFFT Inverse Fast Fourier Transform

LAA Licensed Assisted Access

LB Licensed Band

LTE-A Long Term Evolution-Advanced

LTE-U Long Term Evolution-Unlicensed

MNO Mobile Network Operator

MPD Media Presentation Description

OFDM Orthogonal Frequency Division Multiplexing

OFDMA Orthogonal Frequency Division Multiple Access

PDF Probability Density Function

PFS Proportional Fair Scheduling

QoE Quality of Experience

QoS Quality of Service

QSI Quality Selection Interval

RAT	Radio Access Technology
RB	Resource Block
SI	Scheduling Interval
SNR	Signal to Noise Ratio
SR	Spectrum Refarming
SVC	Scalable Video Coding
TDD	Time Division Duplexing
UE	User Equipment
UB	Unlicensed Band

1 Introduction

Cellular networks are facing the serious problem of spectrum scarcity in recent years. As mobile devices are capable of running applications that demand a considerable amount of bandwidth, e.g. video streaming applications, new challenges rise for Mobile Network Operators (MNO). LTE-Advanced (LTE-A) networks are able to satisfy this vast need of their subscribers for high data rates and as this need increases, the 3rd Generation Partnership Project (3GPP) aims in satisfying it through several enhancements that are proposed for the next releases of LTE-A. One major enhancement is Carrier Aggregation (CA) that has already been employed since Release 10 of the standard and is used to aggregate up to 5 Component Carriers (CC) to a bigger communication channel and thus increase the users' data rates [1]. A number of band combinations for aggregation as well as several types of CA have been proposed ever since but the major problem of spectrum scarcity still remains a challenge. A promising solution that is employed by MNOs is Spectrum Refarming (SR), through which underutilized spectrum reserved for old Radio Access Technologies (RATs) is reassigned to LTE-A. The number of legacy devices that utilize the aforementioned spectrum decreases as they migrate to the newer technology, i.e. LTE-A, so a portion of it can be redistributed to this new technology [2]. This technique however, requires that refarmed spectrum belongs to the same MNO which also has enough of it to satisfy legacy users that still utilize the old RAT.

To this end 3GPP has proposed Licensed Assisted Access (LAA) [3] aiming to the exploitation of Unlicensed Bands (UB) by LTE-A systems. LAA is considered an implementation of the more general concept of LTE-Unlicensed (LTE-U) which entails the exploitation of unlicensed spectrum by LTE systems. The enabling technology behind the proposal of LAA is CA, the only difference being that CA is now between a CC that belongs to the MNO's Licensed Band (LB) and possibly others that belong to an UB that potentially several other devices access, thus creating interference and medium access problems. These problems imply that the additional spectrum provided by UBs may not always be exploitable for LTE-A communications because LAA should guarantee that UB users continue to utilize the UB spectrum (almost) unaffected.

A video streaming application is a type of service that can be greatly improved with the adoption of LAA. The abundance of video content that exists nowadays in addition to the users' increased number of requests and the demand of such high quality content, makes video streaming a typical example of a bandwidth demanding application. Adaptive video streaming protocols such as Dynamic Adaptive Streaming over HTTP (DASH) [4] try to efficiently deliver video data to mobile users by estimating the wireless channel's throughput performance and delivering the video file in segments of a quality level that is proportional to the link's throughput. This is due to the fact that the higher the video quality, the higher the encoding rate of the segments is. Consequently the required data rate that the user must achieve in order to finally watch a video under high Quality of Service (QoS) standards is increased. The actual throughput of the link can greatly vary over time due to the unstable nature of the wireless channel, making its estimation a difficult task, so ideally a solid implementation of DASH should try to fill up the video playback buffer when a good channel quality occurs, in order to cope with a probable bad channel quality that may follow. In addition to the channel quality, the unpredictable availability of resources in a LAA system makes the problem of adaptive video streaming even more challenging.

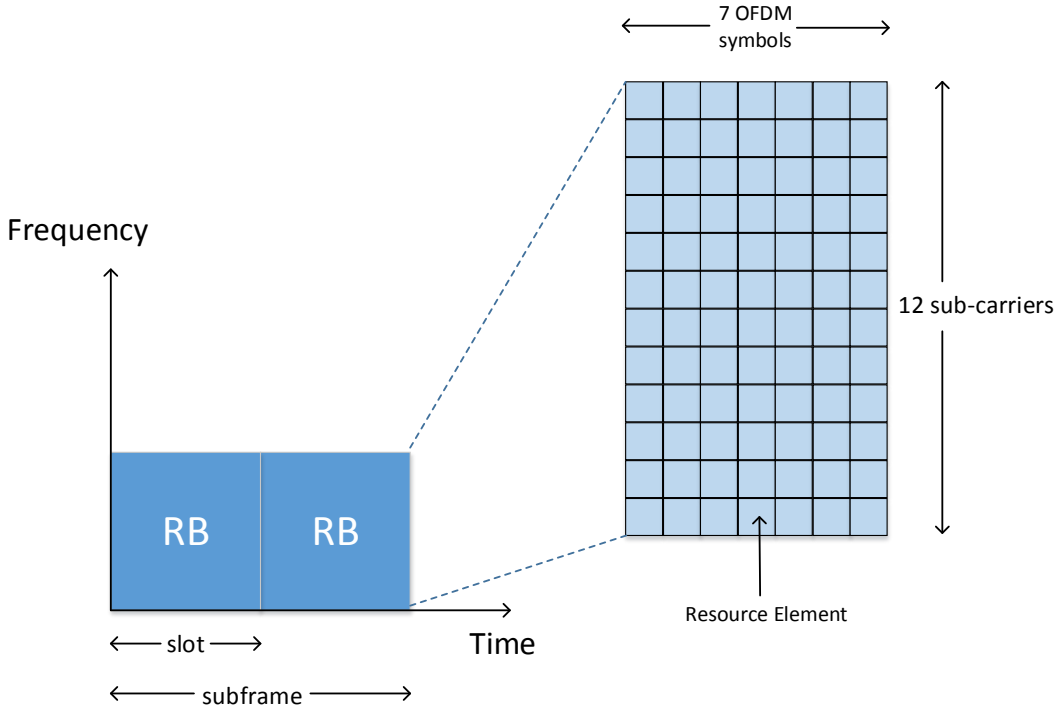

Figure 1: LTE Resource Grid structure.

1.1 LTE basics

LTE-A cellular networks are increasingly deployed worldwide due to their great performance capabilities making them ideal for high QoS applications such as video streaming. Orthogonal Frequency Division Multiple Access technology is utilized to schedule blocks of sub-carriers from the entire transmission bandwidth to multiple users. Each block of sub-carriers is called Resource Block (RB) and consists of 12 sub-carriers spaced at 15 KHz each [5]. This means that the overall bandwidth of a RB is 180 KHz. The number of RBs available for scheduling depends on the total bandwidth of the system. Time is divided into frames, each one lasting for 10ms. Frames are divided to 10 1ms sub-frames, which in turn consist of 2 0.5ms time slots. Each time slot carries 7 OFDM modulated symbols. This entire organization is depicted in Figure 1. Concerning duplexing, both Frequency Division Duplexing (FDD) and Time Division Duplexing (TDD) are supported. For FDD a different frequency is used for the uplink so that downlink and uplink transmissions can occur at the same time. In TDD however, the same spectrum that is used for downlink is also used for uplink, resulting in a number of 7 different frame configurations that indicate the exact sub-frames that are used for downlink and uplink.

1.2 Carrier Aggregation in LTE-Advanced

Carrier Aggregation is a technology used in LTE-A to increase the transmission bandwidth and thus achieve the target data rates set for 4G cellular communications. LTE supports the following bandwidths per CC: 1.4,3,5,10,15,20 MHz [6]. Several CCs of

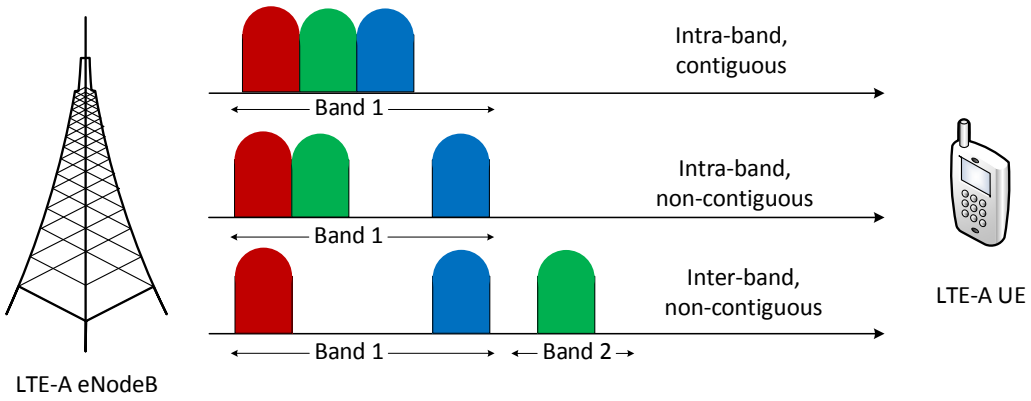

Figure 2: Types of Carrier Aggregation.

possibly different bandwidths can be used by an eNodeB to allocate resources on multiple CCs, provided that the User Equipment (UE) is CA enabled and can decode a CA signal. Three types of CA exist depending on the availability of carriers that determine the Physical Layer architecture of the communicating pair and are displayed in Figure 2. In intra-band contiguous CA all aggregated CCs belong to the same frequency band and occupy contiguous carrier center frequencies. In intra-band non-contiguous CA all CCs belong to the same frequency band, however not all of them employ contiguous carrier positions wherein the band. Finally, in inter-band non-contiguous CA, the CCs belong to different bands and thus are not contiguous in frequency.

1.3 Licensed Assisted Access

3GPP has introduced the concept of utilizing UB through CA to improve users' data rate with the so called Licensed Assisted Access [3]. A LAA system employs at least 2 CCs one of which is in a LB and the rest of CCs are in an UB where other systems may operate. This requires an intra-band non-contiguous CA implementation at the eNodeB since the aggregated CCs belong to different bands. The 5 GHz band is mainly considered for LAA due to the big amount of available spectrum but there are several restrictions in the utilization of the band in order to avoid interfering with other systems. Furthermore, each country has defined different regulations about the utilization of the sub-bands that constitute the 5 GHz band. To this end, and since 3GPP aims in a global application of LAA, the frequency chunk that is expected to be utilized by LAA systems is supposed to be accessed mainly by WiFi users. LAA systems need to incorporate a series of functionalities that will ensure the smooth operation of such systems. These main functionalities defined in [3] have been studied in [7] and are summarized as follows:

- Listen Before Talk. Perform a Clear Channel Assessment (CCA) prior to LTE transmission to ensure an idle channel that will not interfere to other systems' transmission.

- Carrier Selection. The aggregated CC in the UB should be on a low traffic/interference condition concerning the activity of other systems.
- Discontinuous Transmission. LAA transmissions cannot occupy the UB indefinitely, and give the chance to other systems that compete for the same channel to transmit their data.
- Transmit Power Control. Regulations in different regions of the world impose a maximum transmit power level in unlicensed bands.

It is clear that all the above functionalities and their implementation puts the availability of UB spectrum under question. A LAA system may schedule its users in multiple CCs through CA but whether the UB CC will be available or not, highly depends on other systems' activity and implementation of the aforementioned functionalities.

1.4 Adaptive video streaming

Adaptive video streaming is a video streaming technique, where a client requests video files stored in a video server in different quality levels (encoding rates). The file is transfered in chunks of specific duration and depending on the data rate the client can achieve, it decides the quality level of each chunk so that it is encoded in as high rate as possible, while ensuring it can be received in time for playback. Obviously, the higher the encoding rate of a chunk, the longer it is in size and consequently it requires a higher data rate link to be delivered in time. Figure 3 displays an example of adaptive video streaming. The UEs notify the eNodeB for the desired quality of the next chunk and the eNodeB makes the respective chunk request to the Web Server, which in turn transmits the chunk of the desired quality. Then the eNodeB is responsible for scheduling resources to the UEs in order to transmit the chunk before a buffer underrun event occurs, i.e. the chunk has not yet been received and the playback buffer is empty.

The client acquires information about the available video qualities through the Media Presentation Description (MPD) file. This file contains the encoding rate and resolution of each quality level that is stored at the server. Then the client, according to the current achievable bit-rate and the situation of the playback buffer, decides which quality level to request for the next segment.

1.5 Related works and motivation

Video delivery in LTE based cellular networks has been extensively studied in recent years. In [8] the energy efficient delivery of DASH video segments is studied in a LTE heterogeneous network. The problems of user association and resource allocation are studied jointly so that the network's users can download video files with as much high quality as possible, while also trying to minimize the total power consumption accounting for radio transmission and network backhaul power. In [9] the concept of adaptive video streaming with Scalable Video Coding (SVC) over a shared frequency band is studied. The dynamics of the unlicensed users are modeled by a Markov decision process and are incorporated to the system in order to make optimal decisions about the quality of the future segment requests. In [10], [11] the problem of resource

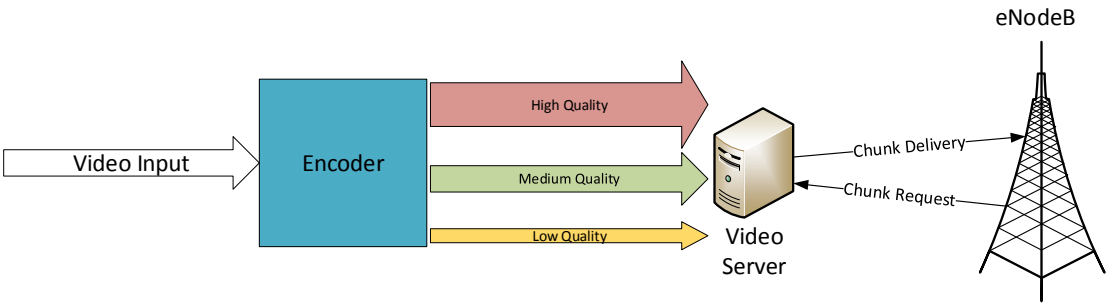

Figure 3: Adaptive video streaming illustration.

allocation in LTE CA systems is considered. The solution involves rate allocation of multiple CCs to the UEs of the network by maximizing logarithmic and sigmoidal like utility functions that represent user satisfaction. In addition, a distributed version of the resource allocation algorithm is presented, that is based on UE bidding for resources process. In [12] a scheduling framework (AVIS) for adaptive video streaming over cellular networks is presented. The authors propose a gateway level architecture for AVIS by implementing it in two entities. The first one is responsible for deciding the encoding rate for each user, while the second one allocates resources in a way that the users' average data rate is kept stable so that segments are downloaded on time. While this framework is in many ways similar to the one proposed in this work, it lacks exploitation of UE buffer status as well as unlicensed band availability information. In [13] resource allocation is achieved by an interference mitigation scheme for Heterogeneous Networks. A stochastic scheduling algorithm is applied to schedule resources probabilistically, that is also observed to increase femtocell capacity. The works in [14], [15] study admission/congestion control and transmission scheduling in small cell networks for adaptive video streaming. More specifically in [14], a network utility maximization problem is formulated in order to keep transmission queues of helper nodes stable. The admission control policy problem is tackled by choosing the helper node as the one with the smallest queue backlog. Transmission scheduling requires the maximization of sum rates with the queue backlogs serving as weights. Furthermore in [15], an algorithm is proposed that calculates the pre-buffering and re-buffering time for each user so that they can experience a smooth streaming service without buffer under-run events. A similar work is also presented in [16], where users are able to download from a number of base stations and decide which of them is better to serve them.

The main contribution of this work is that it combines the adaptive video streaming framework of DASH with opportunistic scheduling due to the existence of the unlicensed CC under the LAA concept. Although adaptive video streaming frameworks have been extensively studied before, there is no work studying the application of adaptive video streaming over a LAA system. Furthermore, since LAA is key for future 5G networks, the increased data rate performance it can offer is extremely important for applications that require high data rates such as video streaming, thus motivating this work.

2 System model

We consider a LTE-A eNodeB with LAA capabilities, i.e. the functionalities described in Section 1.3, enabling it to monitor traffic in one or more unlicensed band CCs. At each scheduling interval t, the eNodeB can employ CA to schedule resources from a licensed primary CC and an unlicensed secondary CC, both in FDD mode, each one of bandwidth W_L and W_U respectively. Depending on the values of W_L and W_U a number of RBs M_L and M_U are available for scheduling. Each RB consists of 12 subcarriers spaced at 15 KHz providing a total bandwidth of $W = 180\ KHz$ per RB. A set of UEs $\mathcal{K}$ exists in the area of the eNodeB and each user $k \in \mathcal{K}$ requests DASH video files. A visual representation of the system is depicted in Figure 4. UEs are in the coverage area of the licensed carrier (light blue color) and the unlicensed one (dark blue color), where WiFi systems also operate. The coverage area of the unlicensed carrier is typically smaller because it is centered at a higher frequency.

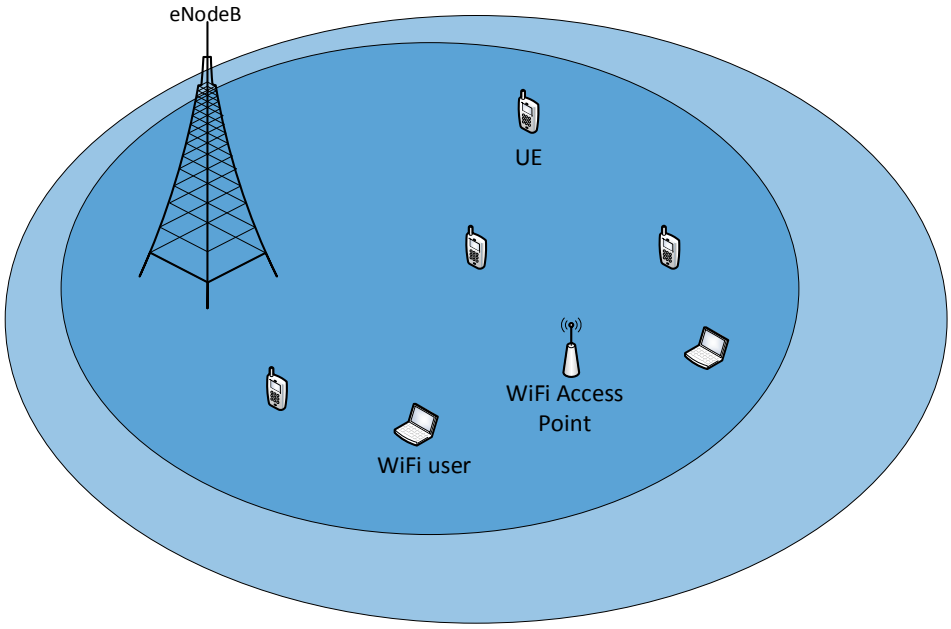

Figure 4: The considered network topology.

2.1 Video streaming

The encoding rate of each video chunk is typically decided by the UE approximately every 10 seconds (each chunk no matter its size corresponds to 10 seconds video duration). We denote this time period as the Quality Selection Interval (QSI). The eNodeB decides the quality for each UE depending on channel quality, secondary CC availability and buffer status of the UEs. Since the QSI is relatively long, it is reasonable to assume that each UE can communicate all necessary information such as buffer status

Quality	Resolution	Bitrate (Kbps)
360p	640×360	1000
480p	848×480	2500
720p	1280×720	5000
1080p	1920×1080	8000
1440p	2560×1440	10000
2160p	3840×2160	35000

Table 1: Quality level encoding rates.

and channel state to the eNodeB through the uplink channel. Channel quality and secondary CC availability are averages observed by the UEs and the eNodeB respectively. The buffer status is the remaining video playback time that is already stored to each UE's playback buffer. We denote the set of available encoding rate levels for the video file requested by UE k as:

$$\mathcal{D}^k = \{D_1^k, D_2^k, ..., D_L^k\} \tag{1}$$

where D_i^k is the encoding rate of the video quality i in bits per second. Furthermore, we assume that the encoding rate (i.e. video quality) increases with the index i, so that $D_i^k > D_{i-1}^k$. L is the number of encoding rates in which the requested video is available by the server. An example of recommended encoding rates for different quality levels is provided in Table 1, where the frame rate is 30 frames per second and the aspect ratio, i.e. the ratio of row to column pixels, is 16:9 which are typical values.

2.2 Unlicensed band traffic estimation

The eNodeB is required to monitor the activity of the UB in order to coexist harmoniously with the deployed UB systems as proposed by 3GPPP [3]. This requires the existence of an energy detector that is capable of collecting samples periodically, and determining whether a signal is present at the desired channel. The statistics of the samples collected by the energy detector can be used to provide an estimate on the probability that the channel is idle, and thus an LTE transmission can take place.

The energy detector collects samples of the unlicensed band and is able to determine whether a transmission takes place or not based on an energy threshold. This energy threshold is typically at -82 dBm for the Carrier Sensing mechanism of WiFi. In [17] however, it is stated that the energy detection level that is used for LAA is set higher at -62 dBm possibly interrupting WiFi transmissions. Thus, the activity of WiFi stations in the unlicensed band can be estimated by measuring the number of samples resulting in busy medium versus the total number of samples collected over a specified time period. So, by defining N_1 to be the number of samples where the detected energy was above the energy threshold, and as N_2 the respective value for below the energy threshold detections we have:

$$P_{on} = \frac{N_1}{N_1 + N_2} \tag{2}$$

being the probability that the unlicensed band is occupied and

$$P_{off} = \frac{N_2}{N_1 + N_2} \tag{3}$$

the probability that the unlicensed band is idle at some random time instance.

The works of Bianchi [18], [19] provide a solid mathematical framework for modeling WiFi users' channel access probability using a discrete time Markov chain are considered, in order to calculate P_{off} under several realistic scenarios. Particularly in [19], the probability that a WiFi station transmits at a random slot is given as:

$$\gamma = \frac{2(1-2p)}{(1-2p)(Win+1) + pWin(1-(2p)^i)} \tag{4}$$

where Win is the minimum backoff window of 802.11 Distributed Coordination Function (DCF), p is the probability that a transmitted packet collides and i is the maximum number of times the backoff window is doubled after consecutive packet collisions. Assuming a number of n stations want to transmit a packet during a slot, p is given by:

$$p = 1 - (1-\gamma)^{n-1} \tag{5}$$

since at least one of the remaining $n-1$ stations should also transmit so that a collision occurs. By solving (4) and (5) we obtain the values of γ and p and the probability p_{tx} that during a random WiFi slot there is at least one transmission that can be detected by the LAA eNodeB (perfect detection is assumed) is given by :

$$p_{tx} = 1 - (1-\gamma)^n \tag{6}$$

We define Ψ to be the random variable that represents the number of consecutive idle slots between two WiFi transmissions. Then the mean of Ψ is given by:

$$\mathbb{E}\{\Psi\} = \frac{1}{p_{tx}} - 1 \tag{7}$$

where $\mathbb{E}\{\cdot\}$ denotes the expectation of a random variable. One more thing is required for the calculation of P_{off}. That is the average duration of a packet transmission in WiFi slots, since $\mathbb{E}\{\Psi\}$ is also calculated in WiFi slots. Assuming that this value is known as $\mathbb{E}\{P\}$ then P_{off} is calculated as:

$$P_{off} = \frac{\mathbb{E}\{\Psi\}}{\mathbb{E}\{\Psi\} + \mathbb{E}\{P\}} \tag{8}$$

P_{off} is therefore a function of Win, i, n and $\mathbb{E}\{P\}$. From these parameters only n and $\mathbb{E}\{P\}$ are considered to vary during time and can affect the performance of our LAA system. Of course these values are unknown to the eNodeB which in practice will perform energy detection based on (3). This model however is required to simulate realistic WiFi traffic scenarios for the system's performance analysis that will follow.

2.3 Solution approach

After quality selection decisions are made, resource allocation is handled by the eNodeB every 10 milliseconds, i.e. the duration of one LTE frame. RBs are scheduled to the

UEs depending on their recent Channel State Information (CSI) as well as the current availability of the secondary CC. We denote this 10 ms resource allocation duration as Scheduling Interval (SI). Note that each QSI interval consists of 1000 SIs. The notation concerning the association of network actions and time is displayed in Figure 5. In the following two sections the problems of quality selection and resource allocation are analyzed under the described setting.

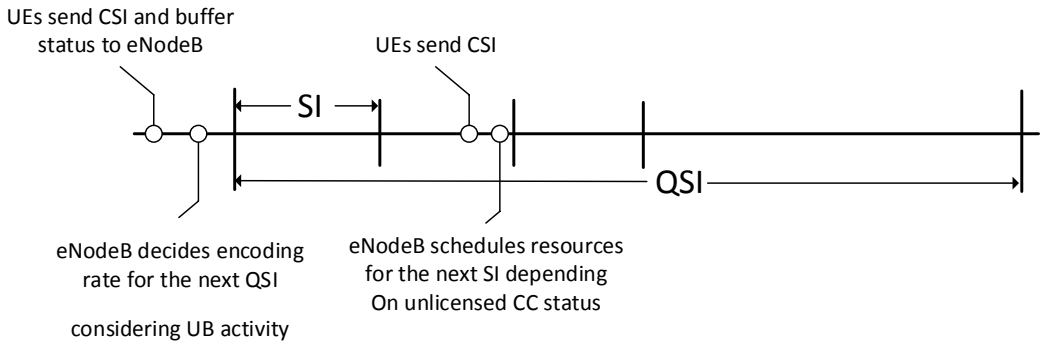

Figure 5: Quality selection and resource allocation decision timeline.

3 Quality Selection

In order for the eNodeB to decide the video quality of the chunks to be delivered for the next QSI, assuming that the current QSI is T, the following information is required. Each UE $k \in \mathcal{K}$ reports their average Signal to Noise Ratio (SNR) experienced of QSI T. These values are denoted by $SNR_L^k(T)$ and $SNR_U^k(T)$ for the licensed and unlicensed CCs respectively. In addition, the UEs should provide their playback buffer status.

3.1 Buffer dynamics modeling

For the UE buffer we assume that during each QSI T, the downloading of the segment to be displayed during the next QSI $T+1$ occurs. The duration of buffered video at QSI T for UE k is given by $B^k(T)$ and is updated for the next QSI as:

$$B^k(T+1) = \begin{cases} \frac{S^k(T)}{R^k(T)}, & \text{if } B^k(T) = 10 \\ B^k(T) + \frac{S^k(T)}{R^k(T)}, & \text{if } B^k(T) + \frac{S^k(T)}{R^k(T)} < 10 \\ B^k(T) + \frac{S^k(T)}{R^k(T)} - 10, & \text{if } B^k(T) + \frac{S^k(T)}{R^k(T)} \geq 10 \end{cases} \tag{9}$$

where $B^k(T)$ is the video duration in seconds stored at QSI T and $\frac{S^k(T)}{R^k(T)}$ is the video duration downloaded at QSI T. $S^k(T)$ denotes the size of the segment(s) in bits to be delivered during QSI T, while $R^k(T)$ denotes the average download rate during QSI T. Differentiating from the work in [20], we assume that at the beginning of each QSI T, the buffer empties by the amount of 10 seconds if the video segment of the previous QSI $T-1$ has been downloaded. If that is not the case, the buffer occupancy at the beginning of QSI T is $\frac{S^k(T)}{R^k(T)}$, which is less than 10 seconds and the completion of the segment download occurs at some point during the next QSI. To help further understand the buffer dynamics an example is illustrated in Figure 6.

Suppose that at the beginning of QSI 1 the buffer contains the first 10-second segment which is downloaded before playback starts. Immediately it is delivered to the application layer and downloading of the next segment begins filling the buffer again. Notice how the slope of the buffer status shows the rate at which the segment is downloaded. For the first two QSIs everything runs smoothly and segments are downloaded on time. This is captured by the first leg of equation (9) where for example $B^k(2) = \frac{S^k(2)}{R^k(2)} = 10$ and thus $B^k(3) = 10$. However, at the third QSI the segment to be delivered to the application at QSI 4 has not been downloaded until QSI 4 begins so that we have $B^k(4) = \frac{S^k(3)}{R^k(3)} < 10$. Since $B^k(4) < 10$, in order to calculate $B^k(5)$ we use the second or third leg of equation (9) depending on if the segment was finally downloaded and delivered to the application during QSI 4. Indeed, since $B^k(4) + \frac{S^k(4)}{R^k(4)} > 10$ we get that the previous segment was downloaded at some point during QSI 4 and the buffer is updated as: $B^k(5) = B^k(4) + \frac{S^k(4)}{R^k(4)} - 10$. The period from the beginning of QSI 4 until the segment is delivered and the buffer empties is the buffering duration when the application layer buffer awaits a segment delivery and the video freezes. The same thing occurs during QSI 5 but with less buffering duration. Finally there is one last possibility for the next QSI buffer status update that is not captured in Figure 6 and that is the second leg of equation (9). Under this case, the segment that had not been downloaded at some QSI T, was still not

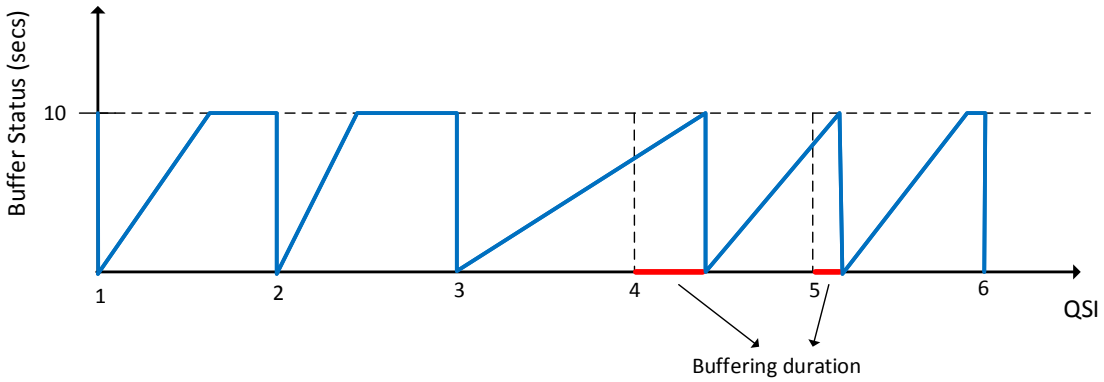

Figure 6: Buffer dynamics modeling example.

received by the end of the next QSI $T+1$ (i.e. $B^k(T) + \frac{S^k(T)}{R^k(T)} < 10$). If this happens the buffering duration lasts for the entire QSI T. However, this case is extremely rare, since additional weight will be given to the scheduling of users not able to download their segments fast enough, as will be shown in the next section.

3.2 Utility maximization for video quality selection

The estimated availability of the unlicensed CC is based on the WiFi users' activity, as sensed by the eNodeB, possibly by energy detection techniques [7], [17]. The probability that the unlicensed CC is idle for QSI T is denoted by P_{off} and is re-evaluated periodically by the eNodeB. The download rate of UE $k \in \mathcal{K}$ that benefits from both licensed and unlicensed CCs at QSI T is then given by:

$$R^k(T) = R^k_L(T) + P_{off} R^k_U(T) \tag{10}$$

In order to calculate $R^k(T)$, one has to perform resource allocation in both licensed and unlicensed CCs by accounting the UEs' SNR values and buffer status, as well as the expected availability of the unlicensed CC. The resulted $R^k(T)$ can serve as an upper bound on the encoding rate of the chunk to be delivered to user k at QSI T. The data rates gained by the licensed and unlicensed carriers are defined as:

$$R^k_L(T) = x^k_L M_L W \log(1 + SNR^k_L(T)) \tag{11}$$

and

$$R^k_U(T) = x^k_U M_U W \log(1 + SNR^k_U(T)) \tag{12}$$

where $x^k_L(T)$ and $x^k_U(T)$ represent the portion of the M_L and M_U sets of resource blocks to be allocated to UE k from the licensed and unlicensed carriers respectively. Therefore we have:

$$x^k_L(T) \in [0,1], \ \forall k \in \mathcal{K} \tag{13}$$

and

$$x_U^k(T) \in [0,1],\ \forall k \in \mathcal{K} \tag{14}$$

Thus, we can define a utility function for each UE k based on $R^k(T)$ and $B^k(T)$, in order to decide the resource allocation that will provide a certain quality selection decision for QSI T as follows:

$$U^k(T) = \log(R^k(T) + \alpha B^k(T)) \tag{15}$$

where α is a biasing factor that will affect the impact of the UEs' buffer status on resource allocation and therefore on quality selection decisions. Notice how the buffer status affects rate allocation. Due to the logarithmic function, UEs with smaller buffer status will tend to be allocated more resources towards balancing their buffer status and will thus manage to download their segment on time. Equation (15) is a concave function with respect to $x_L^k(T)$ and $x_U^k(T)$ so we can formulate the sum utility maximization problem $\boldsymbol{P_1}$ defined as:

$$\boldsymbol{P_1} : \max_{\boldsymbol{x_L}, \boldsymbol{x_U}} \sum_{k \in \mathcal{K}} U^k(T) \tag{16}$$

subject to:

$$\sum_{k \in \mathcal{K}} x_L^k(T) = 1 \tag{17}$$

$$\sum_{k \in \mathcal{K}} x_U^k(T) = 1 \tag{18}$$

where $\boldsymbol{x_L}$ contains variables $x_L^k(T),\ \forall k \in \mathcal{K}$, $\boldsymbol{x_U}$ contains variables $x_U^k(T),\ \forall k \in \mathcal{K}$.

The above problem can be solved with standard convex optimization techniques. Let us define the augmented Lagrangian function by embedding the constraints (17) and (18) in the objective function:

$$\begin{aligned} L(\boldsymbol{x_L}, \boldsymbol{x_U}, \lambda, \mu) = I_k U^k(T) - \lambda(I_k x_L^k(T) - 1) - \mu(I_k x_U^k(T) - 1) - \\ \frac{\rho}{2}(I_k x_L^k(T) - 1)^2 - \frac{\rho}{2}(I_k x_U^k(T) - 1)^2 \end{aligned} \tag{19}$$

where λ, μ are Lagrangian multipliers for each of the two constraints, I_k is a unitary row vector of length $|\mathcal{K}|$ and $\rho > 0$.

In order to maximize L we perform the Alternating Direction Method of Multipliers (ADMM) [21], which involves optimizing L over each variable separately at each iteration τ. Formally we have:

$$\boldsymbol{x_L}^{\tau+1} := \arg\max_{\boldsymbol{x_L}} L(\boldsymbol{x_L}, \boldsymbol{x_U}^{\tau}, \lambda^{\tau}, \mu^{\tau}) \tag{20}$$

$$\boldsymbol{x_U}^{\tau+1} := \arg\max_{\boldsymbol{x_U}} L(\boldsymbol{x_L}^{\tau+1}, \boldsymbol{x_U}, \lambda^{\tau}, \mu^{\tau}) \tag{21}$$

$$\lambda^{\tau+1} := \lambda^{\tau} + \rho(I_k \boldsymbol{x_L}^{\tau+1} - 1) + \rho(I_k \boldsymbol{x_U}^{\tau+1} - 1) \tag{22}$$

$$\mu^{\tau+1} := \mu^{\tau} + \rho(I_k \boldsymbol{x}_{\boldsymbol{L}}^{\tau+1} - 1) + \rho(I_k \boldsymbol{x}_{\boldsymbol{U}}^{\tau+1} - 1) \tag{23}$$

Each of the equations (20) - (21) is solved by setting the partial derivative of L equal to zero, and solving for each variable, which is then used to obtain the Lagrangian variables for iteration $\tau + 1$ through (22) - (23). When ADMM converges, the optimal solution of $\boldsymbol{P_1}$ is found and the achievable data rate of each UE can be calculated. This upper bound of data rate will determine the quality of the video segment to be delivered to the UE as the maximum available from $\mathcal{D}^k$ which does not exceed the achievable data rate $R^k(T)$. The proposed Quality Selection Algorithm is described in Algorithm 1.

Algorithm 1 Quality Selection

```
for each QSI T do
  Require: SNR_L^k(T), SNR_U^k(T), B^k(T), ∀k ∈ K, P_off
  Initialize x_L^0, x_U^0, λ^0, μ^0
  τ ← 0
  repeat
    Calculate x_L^{τ+1}, x_U^{τ+1}, λ^{τ+1}, μ^{τ+1} by eq. (20)-(23)
    τ ← τ + 1
  until Optimal solution found
  for each k ∈ K do
    Calculate R^k(T) by eq. (10)-(12)
    D^k(T) ← max D^k, {D^k(T) ≤ R^k(T)}
  end for
end for
```

The test for convergence is carried out as follows [21]. If $\boldsymbol{x}_{\boldsymbol{L}}^*$, $\boldsymbol{x}_{\boldsymbol{U}}^*$, λ^* and μ^* are the optimal values obtained from the ADMM algorithm, the following conditions must be satisfied:

$$\frac{\partial L(\boldsymbol{x}_{\boldsymbol{L}}^*, \boldsymbol{x}_{\boldsymbol{U}}^*, \lambda^*, \mu^*)}{\partial \boldsymbol{x}_{\boldsymbol{L}}} = 0 \tag{24}$$

$$\frac{\partial L(\boldsymbol{x}_{\boldsymbol{L}}^*, \boldsymbol{x}_{\boldsymbol{U}}^*, \lambda^*, \mu^*)}{\partial \boldsymbol{x}_{\boldsymbol{U}}} = 0 \tag{25}$$

$$I_k \boldsymbol{x}_{\boldsymbol{L}}^* = 1 \tag{26}$$

$$I_k \boldsymbol{x}_{\boldsymbol{U}}^* = 1 \tag{27}$$

Conditions (24) and (25) assure that the solution found is optimal while (26) and (27) that it does not violate the problem's constraints. Proof of the linear convergence of ADMM is provided in [22]. Figure 7 displays the average number of iterations required for ADMM to converge to the optimal solution versus the number of users (K) that clearly affect the number of problem variables and thus the convergence rate of ADMM. It is evident that the increase of iteration steps is linearly proportional to K, which makes the application of Algorithm 1 an efficient solution to the Quality Selection problem $\boldsymbol{P_1}$.

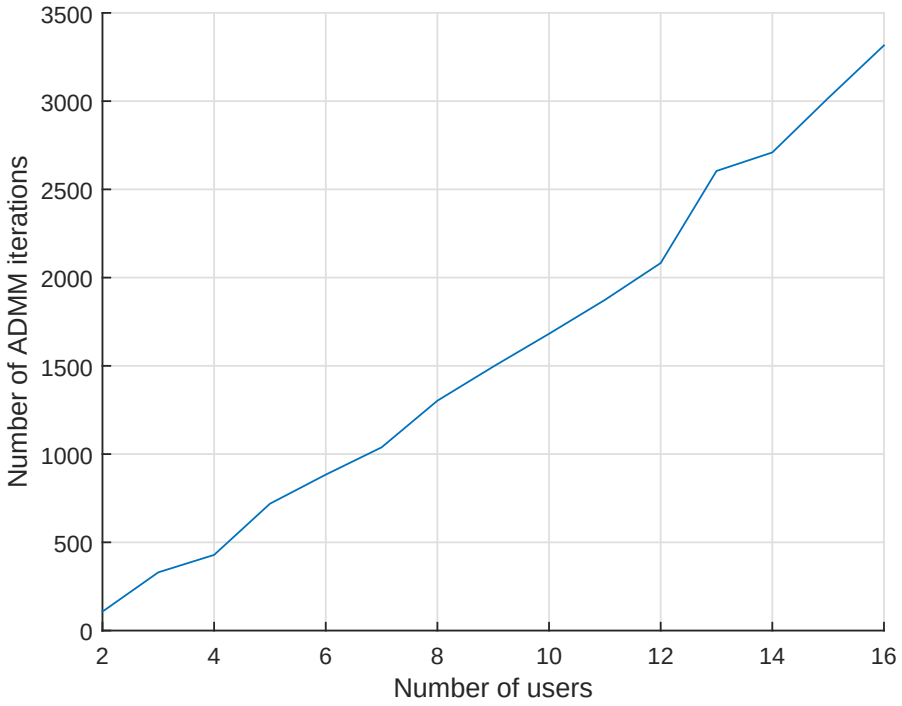

Figure 7: Average ADMM iterations versus number of users.

4 Resource Block scheduling

In this section we propose a RB scheduling policy [23] in both licensed and unlicensed CCs. To deal with resource allocation on a frame basis, a stochastic optimization framework is required [24]. The reason is that each 10-second QSI, consists of 1000 10 ms SIs during which scheduling must ensure that all segments to be transmitted to the UEs at the current QSI are received by the UEs before the end of the QSI. Unfortunately, no information about future channel states and UB availability is known to the eNodeB at each SI and thus a stochastic optimization approach is considered.

4.1 Problem formulation

The approach is based on the assumption that at the beginning of each QSI T, the eNodeB has retrieved the video segments of specific quality from the video server according to the solution proposed in the previous section. Then, a number of K queues hold the segments (one queue per user), and the eNodeB must schedule RBs to UEs at each SI with the goal of emptying all queues by the end of the current QSI. This implies that the segments of QSI T are delivered in time to the UEs so that they can watch freeze-free video content. The number of bits stored at the queues of the eNodeB at each SI t are denoted by $Q^k(t)$. For the first SI the queues are initialized with the segment size of the current QSI T as:

$$Q^k(t=1) = S^k(T),\ \forall k \in \mathcal{K} \tag{28}$$

and are updated for the next SI by:

$$Q^k(t+1) = Q^k(t) - \frac{r^k(t)}{100},\ \forall k \in \mathcal{K} \tag{29}$$

where $r^k(t)$ is the data rate in bits per second experienced by UE k at SI t.

The unlicensed carrier can either be available or unavailable at each SI according to the UB activity as specified by LAA. The auxiliary variable $a_u(t)$ indicates if the unlicensed carrier is available at SI t and thus, unlicensed carrier RBs can be allocated at the specific SI.

$$a_u(t) = \begin{cases} 1, & \text{with probability } P_{off} \\ 0, & \text{with probability } P_{on} \end{cases} \tag{30}$$

The SNR experienced by each UE $k \in \mathcal{K}$ at each SI t is denoted by $SNR_L^k(t)$ and $SNR_U^k(t)$ for the licensed and unlicensed band respectively. The maximum throughput a UE can achieve at the licensed and unlicensed carriers is calculated using the current SNR. We introduce the scheduling variable $y_{mk}(t)$ which is defined as:

$$y_{mk}(t) = \begin{cases} 1, & \text{if RB } m \text{ is allocated to user } k \\ 0, & \text{otherwise} \end{cases} \tag{31}$$

The data rate of UE k at each carrier is then calculated as:

$$r_L^k(t) = \sum_{m \in \mathcal{M}_L} y_{mk}(t) W \log(1 + SNR_L^k(t)) \tag{32}$$

for the licensed carrier and:

$$r_U^k(t) = \sum_{m \in \mathcal{M}_U} y_{mk}(t) W \log(1 + SNR_U^k(t)) \tag{33}$$

for the unlicensed carrier. The total achievable data rate of user k at SI t is then calculated as:

$$r^k(t) = r_L^k(t) + a_u(t) r_U^k(t) \tag{34}$$

The goal is to schedule resources at each SI t so that by the end of the QSI T, each UE k will have downloaded the $S^k(T)$ bits of the video chunk. The problem is formally expressed as:

$$\max_{y_{mk(t)}} \sum_{k \in \mathcal{K}} \sum_{t=1}^{1000} r^k(t) \tag{35}$$

subject to:

$$\sum_{t=1}^{1000} \frac{r^k(t)}{100} \geq S^k(T),\ \forall k \in \mathcal{K} \tag{36}$$

where $\frac{r^k(t)}{100}$ denotes the amount of bits downloaded by user k during SI t, since each SI lasts for 10 milliseconds and $r^k(t)$ is given in bits per second.

4.2 Backlog and Channel Aware Scheduling Policy

The calculation of $y_{mk}(t)\ \forall t \in [1, ..., 1000]$ in (35), requires prior knowledge of $SNR_L^k(t)$, $SNR_U^k(t)$ and $a_u(t)\ \forall t \in [1, ..., 1000]$, which is unavailable at the beginning of QSI T. To tackle this problem we propose a scheduling policy for each SI t. It accounts for the current channel conditions and unlicensed CC availability: $SNR_L^k(t), SNR_U^k(t), a_u(t)$ as well as the users' current backlogs $Q^k(t)$. The proposed algorithm is based on the *max-weight* algorithm of [24] where scheduling decisions are made based on current queue backlogs and channel states without the need of channel probability knowledge.

There are several scheduling policies for LTE systems in the literature [23] both for time and frequency domain scheduling. Leveraging the Orthogonal Frequency Division Multiple Access (OFDMA) technology of LTE we allocate LTE RBs to different UEs at each SI t, applying thus frequency domain scheduling. Under this type of scheduling, a metric function $\delta(m, k)$ is calculated and then each RB m is iteratively allocated to the UE for which the metric function obtains the highest value. Proportional Fair Scheduling (PFS) for example, considers the users instant data rate on each RB as well as the average rate experienced by each user in order to formulate the metric function. In our case however, each UE reports one SNR value for each CC, $SNR_L^k(t), SNR_U^k(t)$ respectively. It is possible though to require sub-band instead of wide-band level feedback reports and thus acquire feedback in the form of $SNR_L^k(m,t), SNR_U^k(m,t)$, where $SNR_L^k(m,t)$ is the SNR experienced by UE k on SI t for RB m of the licensed CC and $SNR_U^k(m,t)$ is the respective value for the unlicensed CC. Whichever the case, $r^k(t)$ can be decomposed to a series of data rates given by the different reported SNRs, whether they differ per RB or they are the same for RBs of the same CC as:

$$r^k(t) = \sum_{m \in \mathcal{M}_L} y_{mk}(t) r_L^k(m,t) + a_u(t) \sum_{m \in \mathcal{M}_U} y_{mk}(t) r_U^k(m,t) \tag{37}$$

where $r_L^k(m,t)$ is given by:

$$r_L^k(m,t) = W \log(1 + SNR_L^k(m,t)) \tag{38}$$

and $r_U^k(m,t)$ by:

$$r_U^k(m,t) = W \log(1 + SNR_U^k(m,t)) \tag{39}$$

which are the data rates in bits per second experienced by UE k on SI t and on RB m if m belongs to the licensed and unlicensed bands respectively. For simplicity in further analysis we assume a wide-band feedback report system and thus equations (38) and (39) reduce to:

$$r_L^k(m,t) = W \log(1 + SNR_L^k(t)) \tag{40}$$

and

$$r_U^k(m,t) = W \log(1 + SNR_U^k(t)) \tag{41}$$

After defining the users' instant data rate per RB we proceed by calculating the average throughput experienced by UE k until SI t as:

$$\overline{r^k}(t) = \frac{\sum_{n=1}^{t} r^k(n)}{t} \tag{42}$$

With standard PFS [23] the metric function $\delta(m,k)$ is calculated as:

$$\delta(m,k) = \begin{cases} \frac{r_L^k(m,t)}{\overline{r^k(t)}}, & \text{if } m \in \mathcal{M}_L \\ \frac{r_U^k(m,t)}{\overline{r^k(t)}}, & \text{if } m \in \mathcal{M}_U \end{cases} \tag{43}$$

It is evident from the form of $\delta(m,k)$ that it is maximized for UEs that experience high instant data rate for RB m and low average data rate, providing thus the proportional fairness characteristic of the metric function. Equation (43) is calculated iteratively for each RB and each UE and RB m is allocated to UE k^* that maximizes $\delta(m,k)$. Formally k^* is given as:

$$k^* = \arg\max_k \delta(m,k),\ \forall m \in \mathcal{M}_L \cup \mathcal{M}_U \tag{44}$$

In our system, proportional fairness is implemented with the logarithmic function in the objective function of the Utility maximization problem $\boldsymbol{P_1}$. At this stage, the designed scheduling policy should empty all backlog queues by the end of each QSI. Since each QSI consists of 1000 SIs the desired scheduling policy must result in: $Q^k(1000) = 0,\ \forall k \in \mathcal{K}$. Driven by the results of the *max-weight* algorithm [24], the scheduling metric is given by the data rate experienced by each UE for the current SI multiplied (weighted) by user's queue backlog and the scheduling decision is based on which UE maximizes the newly defined metric:

$$\delta(m,k) = \begin{cases} Q^k(t) r_L^k(m,t), & \text{if } m \in \mathcal{M}_L \\ Q^k(t) r_U^k(m,t), & \text{if } m \in \mathcal{M}_U \end{cases} \tag{45}$$

Each RB is iteratively assigned to the UE that maximizes (45). Its respective queue backlog is updated and the procedure continues until all RBs are allocated. Next, we present the proposed Backlog and Channel Aware Scheduling Policy (BCASP) in Algorithm 2.

4.3 BCASP analysis

For the remainder of this section we provide an insight about the conditions, under which the proposed BCASP manages to deliver the DASH video segments of a QSI T on time, without the UEs experiencing any buffer under-run events which will result in video freezes.

The max weight based BCASP algorithm utilizes the technique of Lyapunov drift minimization, which is briefly explained next. Consider a system with K queues, each of which stores bits to be transmitted to the K users of the system. Each queue k is supplied with an arrival process that adds a number of $A_k(t)$ bits at each SI t. The system's scheduler must decide which user to serve at each SI t depending on the current queue backlogs $Q^k(t)$ as well as current channel states. Suppose that the number of bits that can be transmitted to user k at SI t and is dependent on the channel state is given by $\frac{r^k(t)}{100}$. The number of bits that are actually transmitted to user k, depending on the scheduler's decision is given by $b_k(t)$ which is defined as:

$$b_k(t) = \begin{cases} \frac{r^k(t)}{100}, & \text{if } k \text{ is scheduled at SI } t \\ 0, & \text{otherwise} \end{cases} \tag{46}$$

Algorithm 2 Backlog and Channel Aware Scheduling Policy

for each SI t **do**
  Require: $SNR_L^k(t), SNR_U^k(t), S^k(T), Q^k(t),\ \forall k \in \mathcal{K},\ a_u(t)$
  for each $m \in \mathcal{M}_L$ **do**
    for each $k \in \mathcal{K}$ **do**
      $\delta(m,k) \leftarrow Q^k(t) r_L^k(m,t)$
    end for
    $k^* \leftarrow \arg\max_k \delta(m,k)$
    $Q^{k^*}(t) \leftarrow Q^{k^*}(t) - \frac{r_L^{k^*}(m,t)}{100}$
  end for
  if $a_u(t) = 1$ **then**
    for each $m \in \mathcal{M}_U$ **do**
      for each $k \in \mathcal{K}$ **do**
        $\delta(m,k) \leftarrow Q^k(t) r_U^k(m,t)$
      end for
      $k^* \leftarrow \arg\max_k \delta(m,k)$
      $Q^{k^*}(t) \leftarrow Q^{k^*}(t) - \frac{r_U^{k^*}(m,t)}{100}$
    end for
  end if
end for

Note that this system matches many characteristics of the Resource Block scheduler described in the beginning of Section 4. The only differences are that the scheduling decision is taken over multiple RBs and the fact that the arrival process is simplified so that there is no periodic increase on backlogs but instead they are increased at the beginning of a QSI with the target of emptying by its end. The queue dynamics from one SI to the next are thus given by:

$$Q^k(t+1) = \max\{Q^k(t) - b_k(t), 0\}, \forall k \in \mathcal{K} \tag{47}$$

To minimize notation, let $\boldsymbol{Q}(t) = [Q^1(t), Q^2(t), ..., Q^K(t)]$ be the vector containing the queue backlogs of all K users of the system. The Lyapunof function is defined as:

$$F(\boldsymbol{Q}(t)) = \frac{1}{2} \sum_{k \in \mathcal{K}} Q^k(t)^2 \tag{48}$$

The upper bound on the change of the Lyapunov function from one slot to the next is:

$$F(\boldsymbol{Q}(t+1)) - F(\boldsymbol{Q}(t)) = \frac{1}{2} \sum_{k \in \mathcal{K}} \left[Q^k(t+1)^2 - Q^k(t)^2 \right] \tag{49}$$

$$= \frac{1}{2} \sum_{k \in \mathcal{K}} \left[\max\{(Q^k(t) - b_k(t))^2, 0\} - Q^k(t)^2 \right] \tag{50}$$

$$\leq \sum_{k \in \mathcal{K}} \frac{b_k(t)^2}{2} - \sum_{k \in \mathcal{K}} Q^k(t) b_k(t) \tag{51}$$

where (51) results from the fact that:

$$\max\{(A-B)^2, 0\} \leq A^2 + B^2 - 2AB \tag{52}$$

The conditional Lyapunov drift for slot t is defined as:

$$\Delta(\boldsymbol{Q}(t)) = \mathbb{E}\left\{F(\boldsymbol{Q}(t+1)) - F(\boldsymbol{Q}(t))|\boldsymbol{Q}(t)\right\} \tag{53}$$

$$\leq \mathbb{E}\left\{\sum_{k\in\mathcal{K}} \frac{b_k(t)^2}{2}|\boldsymbol{Q}(t)\right\} - \mathbb{E}\left\{\sum_{k\in\mathcal{K}} Q^k(t)b_k(t)|\boldsymbol{Q}(t)\right\} \tag{54}$$

The first of the two terms in (54) is independent of $\boldsymbol{Q}(t)$ and is bounded by the value obtained if the scheduler allocates all resources to the user with the highest $r^k(t)$ at SI t. Assuming this value is denoted as C we get:

$$\Delta(\boldsymbol{Q}(t)) \leq C - \mathbb{E}\left\{\sum_{k\in\mathcal{K}} Q^k(t)b_k(t)|\boldsymbol{Q}(t)\right\} \tag{55}$$

Now, in order to minimize $\Delta(\boldsymbol{Q}(t))$, we only need to opportunistically maximize:

$$\mathbb{E}\left\{\sum_{k\in\mathcal{K}} Q^k(t)b_k(t)|\boldsymbol{Q}(t)\right\} \tag{56}$$

which, according to the max weight algorithm in [24], is achieved by making scheduling decisions that maximize $Q^k(t)\frac{r^k(t)}{100}$. Since scheduling decisions are taken for each individual RB of licensed and unlicensed CCs the scheduling metric in (45) complies with the max weight opportunistic scheduling results.

The BCASP is designed to guarantee the stability of the transmission queues in the long term, meaning that all video segments will be downloaded at some point, but it provides no guarantees about the short term delivery of each individual segment during a QSI. Channel conditions may deteriorate for all users and unlicensed carrier utilization by WiFi users may increase dramatically during a 10 second period, allowing fewer access opportunities and decreased data rates. Consequently the eNodeB might not be able to provide the promised data rates so that the UEs can download their segments on time for playback. Is there something we can do to minimize the chances of buffer under-run events? First, the total data rate that the eNodeB must provide is mostly much lower than the one it can provide. Remember that the encoding rate of the segments to be delivered to the UEs is selected as the highest available that does not exceed the UE data rate calculated by equations (10)-(12) after the utility maximization problem is solved. This difference between required and available data rate is in most cases enough to minimize video freezes as will be shown in the next section. In addition, a simple pre-buffering technique is proposed in [16] where the pre-buffering time (the time between the request of the first video segment until playback starts) is calculated as the maximum delay of the transmission queues of the users. This implies that UEs can request more segments before the current segments are downloaded in order to get a head-start in preventing a buffer under-run event due to the factors already mentioned. Although such technique may even eliminate the chances of buffer under-run events, it might also add a significant delay at the beginning of playback which will most certainly degrade Quality of Experience (QoE) for the users.

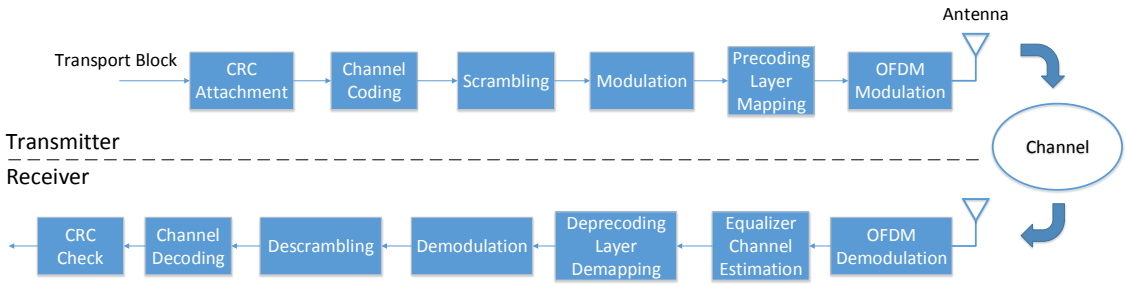

Figure 8: LTE Physical Layer downlink processing chain.

5 Performance evaluation

In this section extensive simulation results on the considered LAA video streaming framework are provided. In the first two sub-sections, the simulation setup is explained both in link level (LTE physical layer) as well as in system level, in order to lay the foundation for the upcoming simulations and the explanation of results in the following sub-section.

5.1 Link level simulation setup

The link level simulation consists of the implementation of LTE Physical Layer in MATLAB in order to obtain performance metrics of the downlink transmission of LTE under different channel conditions and transmission modes. The LTE downlink processing chain is provided in Figure 8. The functionality of the blocks depicted in Figure 8 is briefly described as follows:

- CRC Attachment. Cyclic Redundancy Check (CRC) is used for error detection of the transport blocks. A number of parity bits are attached to the end of the transport block.
- Channel Coding. Turbo coding with a rate of 1/3 is usually employed to protect the transmission of data against channel fading.
- Scrambling. The input bit streams are combined to scrambling sequences in order to produce pseudo-random codewords.
- Modulation. The scrambled bits are used to generate complex valued modulation symbols. The constellations supported by LTE are QPSK, 16QAM and 64QAM in order to provide adaptive modulation.
- Precoding and Layer Mapping. This functionality is employed in case of a multi-antenna transmission scheme. It involves the mapping of the input symbols on each layer for transmission on the available antenna ports.
- OFDM Modulation. Orthogonal Frequency Division Multiplexing modulation is applied by utilizing Inverse Fast Fourier Transform (IFFT) in order to convert a frequency selective fading channel to a number of flat fading orthogonal sub-channels of narrower bandwidth.

Parameter	Value
Bandwidth	1.4,3,5,10,15,20 MHz
Tx/Rx Antennas	1,2,4
Modulation	QPSK, 16QAM, 64QAM
Tx Scheme	Single port, Tx Diversity, Spatial Multiplexing
Channel Model	EPA, EVA, ETU

Table 2: Physical layer simulation setup parameters.

The reverse actions are made at the receiver side with the addition of Channel Estimation which is used to counteract the effects of the wireless channel on the received signal and improve Bit Error Rate (BER). The procedure involves the reception of known pilot symbols in specific Resource Elements of the LTE Resource Grid. The receiver can then estimate the effect of the channel by observing the difference between the known and received pilot symbols.

The simulation environment developed to model the physical layer described above sets system parameters such as the number of frames, bandwidth, transmission scheme etc and passes the OFDM modulated signals through a fading channel for a series of SNR values. The receiver decodes the received signals and the throughput performance of the link is calculated. A complete list of the parameters and possible values supported by the simulator are given in Table 2.

Some typical results provided by the simulator are shown in Figure 9. The duration of the simulations was 100 LTE frames and the channel model used was the Extended Pedestrian A model. Adaptive modulation was employed according to [25]. The figure depicts all supported transmission schemes for different bandwidth and Tx/Rx antenna setups. In the following sections a specific setup will be used for all links in the cellular network in order to provide system level results of the UE scheduling for video streaming application.

5.2 System level simulation setup

Now we describe the details of the LAA system that provides streaming services with the functionalities described in the previous sections. We consider a cell topology like in Figure 10 with a number of K UEs spread uniformly in a 2×2 kilometer area and the LAA enabled eNodeB at the center of it. One licensed and one unlicensed CCs of 20 MHz, each one entailing a number of $M_L = M_U = 100$ RBs are considered. The link level profile of both CCs is assumed to be 4×1 transmit diversity as displayed in Figure 9.

The SNR in dB that each UE k experiences for the licensed CC at each QSI T is given by:

$$SNR_L^k(T) = P_L^{rx}(dBm) - N_0(dBm) \tag{57}$$

where $P_L^{rx}(dBm)$ is the received power in dBm for the licensed CC, and $N_0(dBm)$ is the noise power. In order to calculate $P_L^{rx}(dBm)$, large scale fading with path loss and log-normal shadowing is considered as:

$$P_L^{rx}(dBm) = P^{tx}(dBm) - FSPL_L(dB) - X_s \tag{58}$$

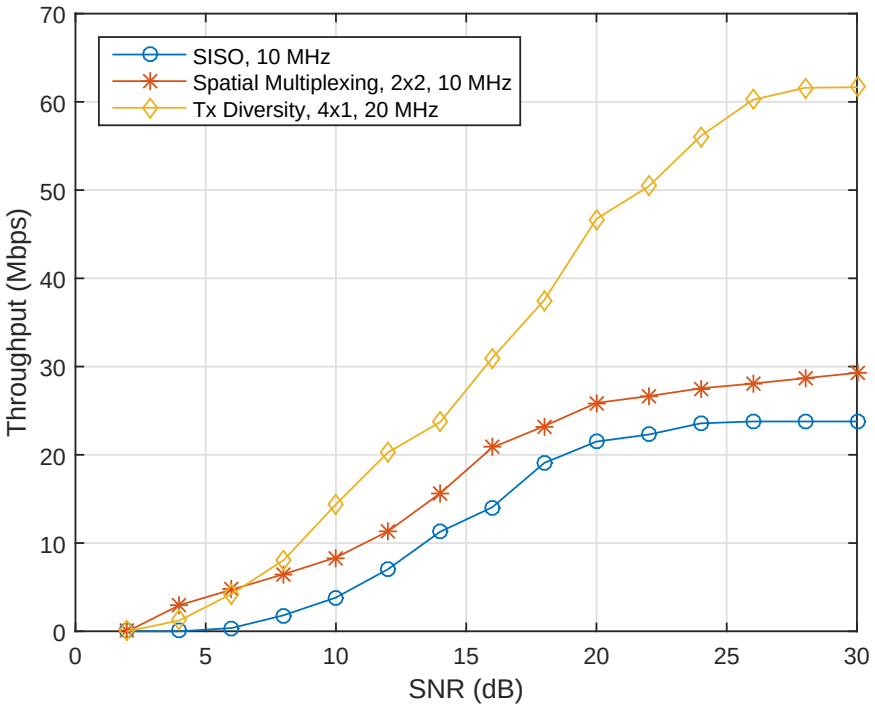

Figure 9: Physical layer throughput versus SNR.

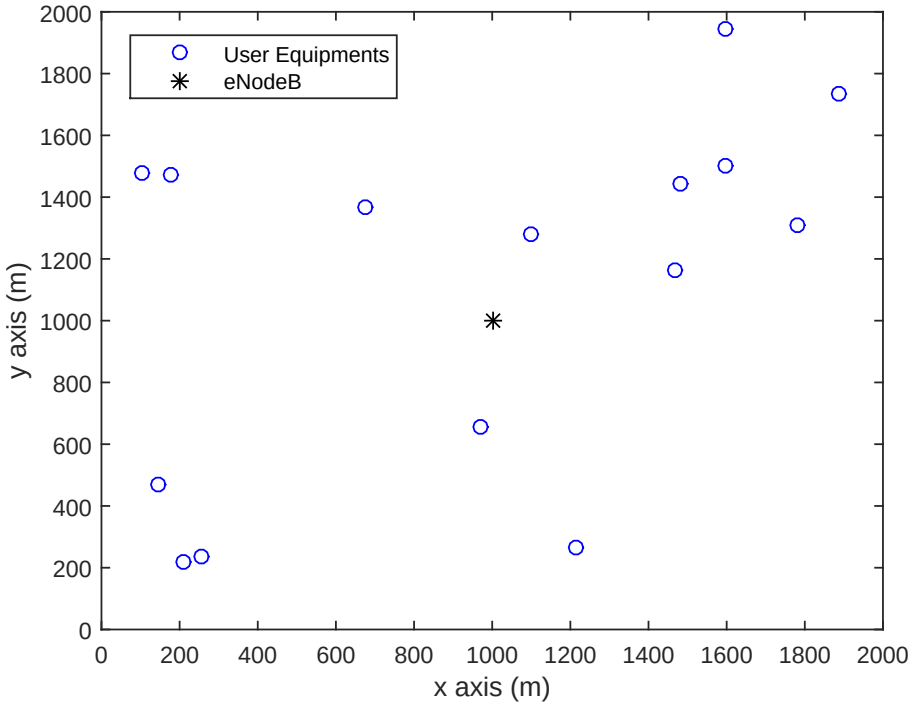

Figure 10: Instance of the network topology for $K = 15$ UEs.

Parameter	Value
W_L/W_U	20 MHz
M_L/M_U	100
f_L/f_U	2.1/5.8 GHz
P^{tx}	43 dBm
N_0	-80 dBm
σ^2	3

Table 3: System level simulation setup parameters.

where $P^{tx}(dBm)$ is the transmit power of the eNodeB in dBm, $FSPL_L(dB)$ is the Free Space Path Loss (FSPL) for the licensed CC in dB and X_s, which accounts for log-normal shadowing, is a Gaussian random variable with $\mathcal{N}(0, \sigma^2)$. $FSPL_L(dB)$ is given by:

$$FSPL_L(dB) = 20(\log_{10}(d) + \log_{10}(f_L) - 7.378) \tag{59}$$

where d is the distance between transmitter and receiver in meters, and f_L is the center frequency of the transmitted signal in the licensed band in Hertz. The unlicensed CC belongs to a different spectrum band and thus FSPL is given by:

$$FSPL_U(dB) = 20(\log_{10}(d) + \log_{10}(f_U) - 7.378) \tag{60}$$

where f_U is the center frequency of the unlicensed CC. Thus, for the unlicensed CC we get that:

$$P_U^{rx}(dBm) = P^{tx}(dBm) - FSPL_U(dB) - X_s \tag{61}$$

and

$$SNR_U^k(T) = P_U^{rx}(dBm) - N_0(dBm) \tag{62}$$

The average SNR experienced during QSI $T-1$ is then used to calculate $SNR_L^k(T)$ and $SNR_U^k(T)$ as:

$$SNR_L^k(T) = \frac{\sum_{t=1}^{1000} SNR_L^k(t)}{1000} \tag{63}$$

and

$$SNR_U^k(T) = \frac{\sum_{t=1}^{1000} SNR_U^k(t)}{1000} \tag{64}$$

respectively.

The SNRs calculated above consider only large scale fading which is supposed to be stable for the duration of the segments download, considering that the UEs are not moving. However, in order to capture small scale fading which occurs even with minor movements of the receiver, we incorporate Rayleigh fading that changes the experienced SNR from one SI to the next. The details of the values used for the parameters of the system level simulation setup are provided in Table 3.

Concerning the video files that the UEs request for downloading, we consider a video file encoded in 6 different quality levels as described in Table 1. Formally we have that:

$$\mathcal{D}^k = \{1000, 2500, 5000, 8000, 10000, 35000\} Kbps, \ \forall k \in \mathcal{K} \tag{65}$$

Moving on to the WiFi system setup, suppose that a random number of n WiFi stations are involved in packet transmissions during each QSI. Each packet is considered to have a fixed size of 1.5 KB and can be transmitted at a set of physical data rates supported by 802.11n that is operational in the 5GHz spectrum. This set $\mathcal{R}_w$ of physical WiFi data rates is as follows:

$$\mathcal{R}_w = \{7.2, 14.4, 21.7, 28.9, 43.3, 57.8, 65, 72.2\} \ Mbps \tag{66}$$

The rate which will be selected derives from WiFi's rate adaptation mechanism and depends on each station's channel conditions. For a fixed packet size one can calculate the transmission duration for each possible data rate and by accounting that each WiFi slot lasts for 9 μs the set of possible transmission durations in number of WiFi slots is given by:

$$\mathcal{T}_w = \{186, 94, 62, 47, 32, 24, 22, 19\} \ slots \tag{67}$$

Assuming that in each WiFi slot there is a number of n WiFi stations that want to transmit a packet, we can calculate P_{off} for each QSI by equations (4)-(8) and (67). However, since the number of competing WiFi stations during a QSI is variable, and in order to simplify WiFi operation during simulation by explicitly manipulating P_{off} so that its effect is evident, we assume that it follows a Gaussian distribution over QSIs as $\mathcal{G}(\mu, \phi^2)$.

5.3 Simulation results

For the remainder of this section, the performance of the proposed quality selection and scheduling algorithms will be tested through indicative simulations performed using the setup described above. These simulations aim to highlight the effect of several network parameters such as the number of users and the unlicensed band availability, on crucial performance metrics such as average segment quality and number of video freezes. The proposed solution that consists of Algorithms 1 and 2, both implemented on the eNodeB are compared to standard PFS and the AVIS framework [12].

With PFS, the eNodeB allocates resources according to the scheduling metric in (43). Each UE experiences an average data rate, according to which it requests the appropriate quality for the next segment using the same rule as in Algorithm 1, i.e. it requests the maximum available segment encoding rate that does not exceed the average experienced data rate. Since PFS is a general solution and is not specifically designed to address the complex problem of adaptive video streaming, the AVIS framework is also employed for comparison. AVIS consists of two entities. The *allocator*, which considers the resource requirements of the UEs and decides the encoding rate of the segments to be delivered to each UE, and the *enforcer* which allocates resources in a similar to PFS manner so that UEs can download the desired segments in time. This framework is very similar to the one proposed in this work with the allocator

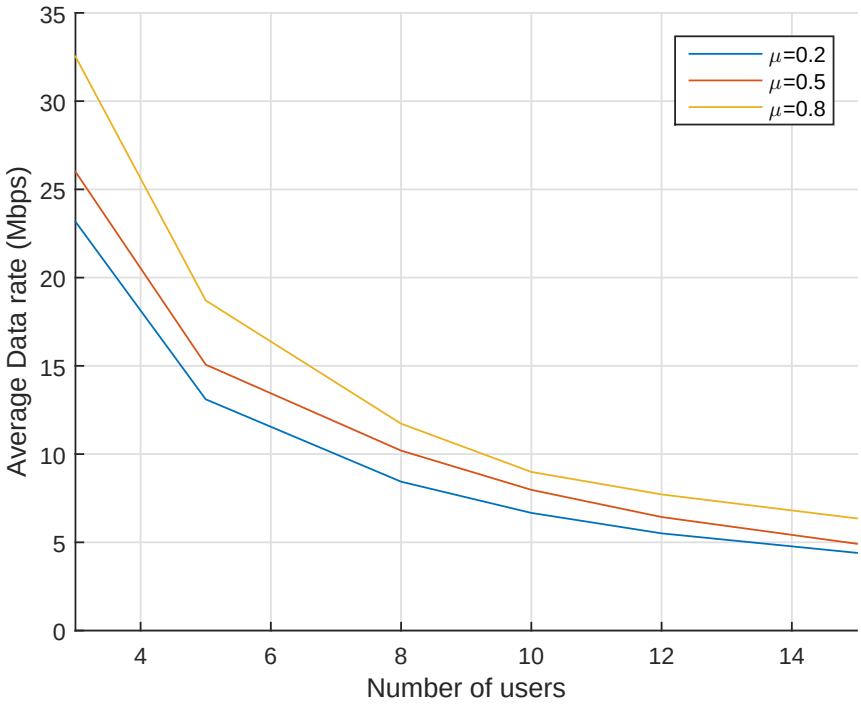

Figure 11: Average data rate vs number of UEs for different cases of unlicensed CC availability.

matching with Algorithm 1 and the enforcer matching with Algorithm 2. Furthermore, the allocator is designed to operate every 10 seconds and the enforcer every 10 milliseconds which is exactly as defined in Figure 5.

5.3.1 Video Segment Quality

The chosen segment quality depends on the data rate that the network can provide to each UE. Furthermore, data rate is a function of the number of UEs associated with the eNodeB as well as the number of available resources which directly links to unlicensed CC availability. To highlight the impact of the above parameters on average data rate, Figure 11 is provided.

Unlicensed CC availability is affected by the mean value μ of P_{off} which varies in Figure 11 while its standard deviation remains fixed at $\phi^2 = 0.1$. As the number of UEs K increases the average data rate decreases since more UEs share the same number of resources. The effect of unlicensed CC utilization can be seen for the three cases of μ depicted in Figure 11. For higher values of μ the unlicensed CC access probability is higher, UEs can be allocated with more resources and thus enjoy higher data rates that can lead to better video quality and/or fewer video freezes. The increased data rate effect is stronger when the number of UEs is small and deteriorates as K increases. However, between different cases of μ the percentile drop in average data rate remains steady. For example, the average data rate is increased by approximately 40% from $\mu = 0.2$ to $\mu = 0.8$ for all values of K. This indicates that even when K is high and the average data rate is relatively small, the increased performance due to unlicensed CC utilization is far from negligible.

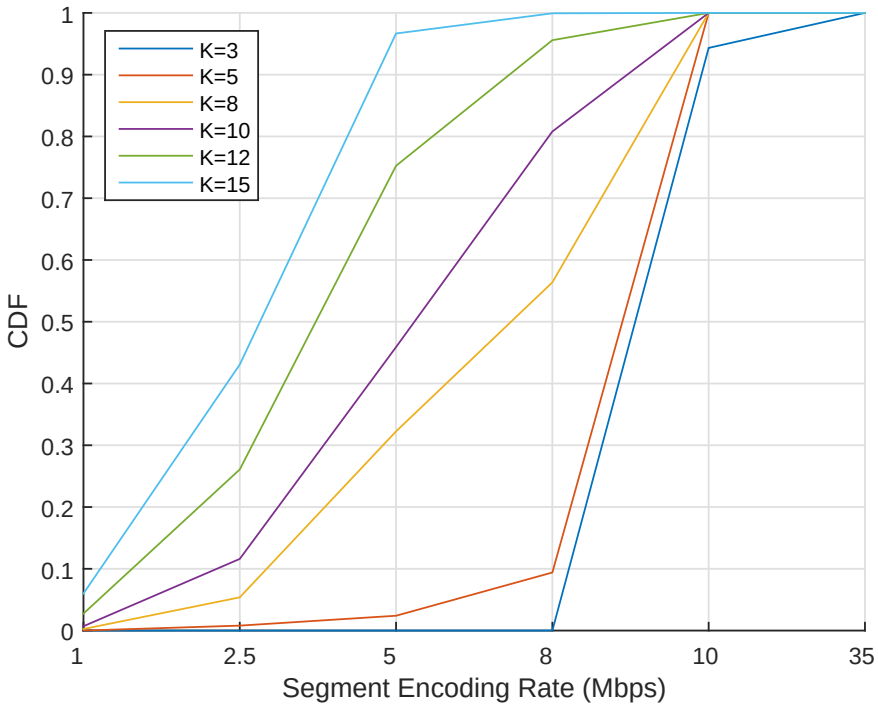

Figure 12: Segment quality CDF for different number of UEs.

As stated before, the average segment quality mainly depends on the number of UEs, since the more UEs are active in the network, the less resources are allocated to each one, and thus they generally experience less throughput which leads the designed quality selection algorithm to choose lower quality segments. Figure 12 displays the Cumulative Distribution Function (CDF) of the different segment qualities of (65) for several cases of number of UEs K and for a number of 100 QSIs. Probability of unlicensed CC availability P_{off} is Gaussian with mean value of $\mu = 0.5$ and standard deviation $\phi^2 = 0.1$.

Intuition is validated through Figure 12 since CDFs for more UEs are above those with less UEs, meaning that they end up with more low quality segments. More specifically for $K = 15$, more than 90% of segments are delivered in the 3 lowest quality levels of 1, 2.5 and 5 Mbps and the rest of them in just 8 Mbps. On the other hand, for $K = 3$, very few segments are delivered in the 4 lowest qualities, while most of them are delivered in 10 Mbps and a small portion of under 10% is even delivered in the highest quality of 35 Mbps.

In addition to the proposed Algorithms 1 and 2 a standard PFS solution as well as the AVIS framework are also considered for comparison. It is expected that PFS will try to provide approximately the same QoS to all UEs and it is interesting to observe the effect of such scheduling policy on segment quality, since there is no knowledge about the unlicensed CC availability to the UEs that request the segments depending on average data rate performance. On the other hand, since AVIS quality selection is implemented on the eNodeB, there is knowledge about the number of available resources but the scheduling is handled just like in PFS. Figure 13 displays the average segment quality CDF for PFS only on a licensed CC, on a licensed plus unlicensed CC

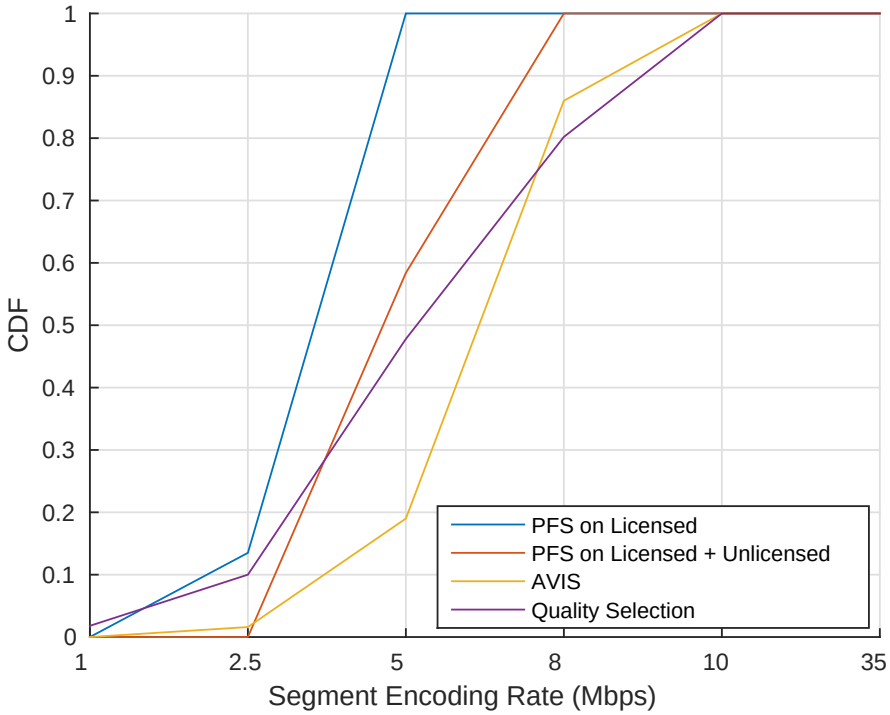

Figure 13: Segment quality CDF for PFS, AVIS and Quality Selection algorithm.

through CA, AVIS, as well as Algorithm 1 solution. The results were obtained for a number of $K = 10$ UEs and a duration of 100 QSIs.

Firstly, by comparing the two PFS results we can see how the utilization of a secondary CC in the unlicensed band has increased average video quality. All users are served by segments the encoding rate of which does not exceed 5 Mbps on a licensed only system. However, when an unlicensed CC is added, a percentage of about 60% is served by the 3 lowest qualities (1,2.5,5 Mbps) while the remaining users experience quality of 8 Mbps segments. When Algorithm 1 is used to select segment quality however, we can see that more UEs are served with segment encoding rates above 5 Mbps compared to the PFS solutions. There is however a small 10% of UEs where the two lowest qualities are chosen while in the respective case of PFS this percentage is 0. In contrast to PFS, Algorithm 1 allocates more resources to UEs with good channel conditions and fewer (thus the worse quality) to the ones experiencing bad channel conditions. In anyway the percentage of UEs with high segment quality is quite bigger than the one of low segment quality proving in a way the usefulness of Algorithm 1. Comparing AVIS to Algorithm 1 however we can see that only about 20% of segments are delivered in the 3 lowest quality levels while for most of them, 8 Mbps encoding rate is chosen. This indicates a better performance that Algorithm 1 but once again we observe that Algorithm 1 provides a larger variety of encoding rates providing more 10 Mbps segments than AVIS. AVIS does not consider UE buffer status and thus never decides to assign a lower quality if the previous segment could not be delivered, thus making the entire framework less adaptive but more aggressive, impacting this way buffer under-run occurrences.

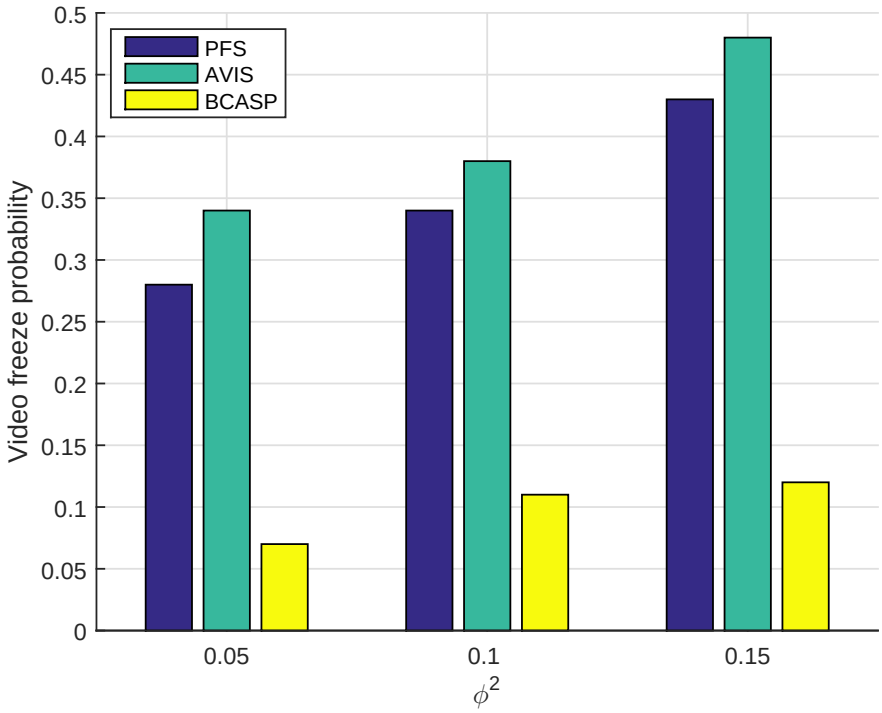

Figure 14: Video freeze probability comparison between PFS, AVIS and BCASP.

5.3.2 Video freezes

The user's QoE is not only determined by the quality of the video displayed. Playback should be smooth with minimum to none interruptions for buffering to ensure maximum viewing experience. BCASP was designed so that the selected video segments will be delivered to the UEs before their playback is due. What makes on time delivery so difficult in our system is the fact the unlicensed CC access probability varies over time, and the system cannot provide stable data rates to the UEs. Thus, Figure 14 displays the average freeze probability for an LAA system where the mean value of P_{off} is $\mu = 0.5$ and its standard deviation takes values from the set [0.05,0.1,0.15]. As the standard deviation increases, the freeze probability is also expected to increase, since unlicensed CC availability varies more through QSIs and so does the average rate, increasing the chance of selecting video qualities that cannot be delivered in the following QSI.

In all tested cases of Figure 14 the dominance of BCASP over PFS and AVIS is evident since the video freeze probability is about 3 times less with BCASP. This is because the eNodeB decides segment quality by considering unlicensed CC traffic dynamics. This helps in making better decisions in contrast to PFS where resources of the unlicensed CC are scheduled upon being available resulting in fluctuating data rates and thus mistaken segment quality selections by the UEs that are unable to see the bigger picture. As for AVIS framework, the aggressive policy of selecting high quality segments impacts freeze probability since the scheduler (PFS) takes no action in order to empty the transmission queues which are anyway difficult to empty due to the quality selection decisions. This results in an even higher freeze probability than PFS.

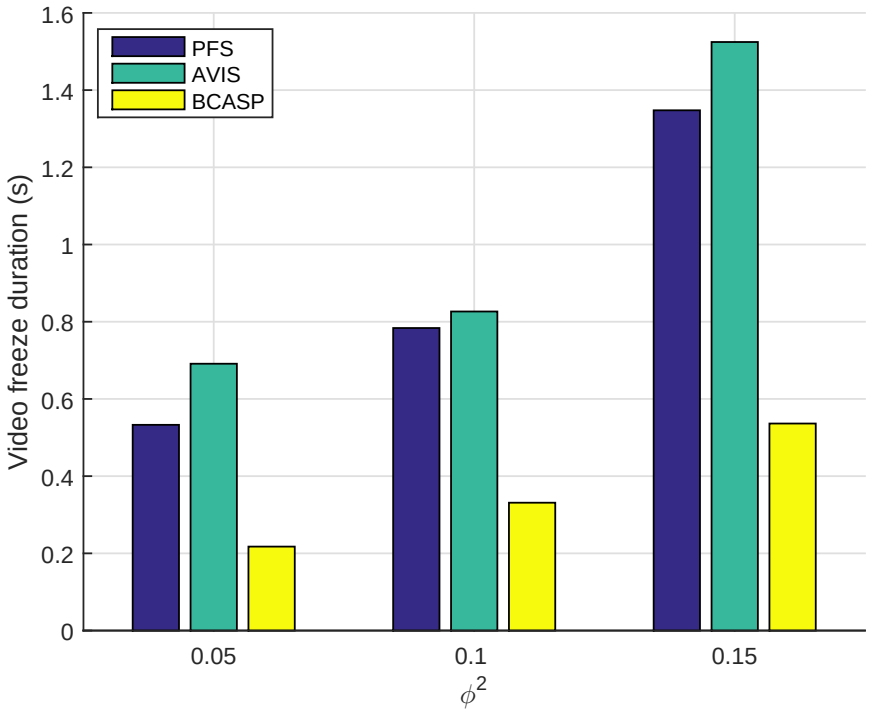

Figure 15: Video freeze duration comparison between PFS, AVIS and BCASP.

The impact of video freezes on QoE lies not only in their probability but also in their duration. A small freeze duration of a couple hundred of milliseconds might even be negligible compared to one that lasts a couple of seconds and maybe more. In Figure 15 we once again compare PFS, AVIS and BCASP in terms of average freeze duration in seconds for the same P_{off} standard deviation values tested in Figure 14.

Once again we observe the superiority of BCASP compared to PFS and AVIS. BCASP schedules resources by considering the queue lengths of the UEs and tries to empty the longest ones. This implies that even in the event of a video freeze and since the scheduler has made its best effort to empty the queues, the remaining duration of the segments not yet delivered is kept at minimum. With PFS on the other hand, there is no such guarantee and the buffering duration is increased. Once again we highlight that the reason why BCASP performs so much better concerning video freezes is that on one hand, the segment qualities have been selected by the eNodeB which has unlicensed CC access probability knowledge, and on the other hand BCASP considers backlogs. AVIS again shows longer freeze duration than both PFS and BCASP since queue backlogs are large and difficult to empty with a PFS approach.

6 Conclusion

This work provides a framework for the application of adaptive video streaming over a LAA system. LAA is a key enabler for 5G radio communications towards increasing subscribers' data rates and satisfying data rate demanding applications such as adaptive video streaming. To this end, the presented analysis exploits the extra unlicensed spectrum by placing segment quality decisions to the eNodeB, which has unlicensed band activity knowledge. Quality selection is accomplished by employing a utility maximization problem at the eNodeB under resource allocation constraints. The problem is solved using the ADMM in linear to the number of users complexity and its solution determines the segment qualities to be delivered to the users for the next interval. Furthermore, a scheduling policy that is ideal for delivering the predetermined amount of payload under specific time constraints is employed. The policy is based on Lyapunov optimization and its advantage is that it tries to schedule resources to the users that not only experience high instant data rate, but also have increased queue backlogs, meaning that there are a lot of bits to be transmitted yet, for the segment download completion. This method is found to optimize performance compared to a standard PFS approach as in two ways. Firstly, the users experience higher quality video with respect to the selected segment encoding rates and secondly the amount and duration of video freezes, due to buffer under-run events that greatly affect viewing experience, is minimized. An existing adaptive video streaming framework that shares many similarities with the one proposed in this work is also employed for comparison. The results indicate that although it generally provides higher segment quality to the users, it is too aggressive resulting in increased buffer under-run probability that causes disturbing and longer video freezes.

References

[1] Qualcomm, "LTE Advanced-evolving and expanding in to new frontiers," tech. rep., August, 2014.

[2] S. Han, Y.-C. Liang, and B.-H. Soong, "Spectrum refarming: A new paradigm of spectrum sharing for cellular networks," in *Global Communications Conference (GLOBECOM), 2014 IEEE*, pp. 893–898, Dec 2014.

[3] 3GPP, TR 36.889 TSG RAN, "Study on Licensed-Assisted Access to unlicensed spectrum," Rel.13 v13.0.0, June 2015.

[4] T. Stockhammer, "Dynamic adaptive streaming over HTTP –: Standards and design principles," in *Proceedings of the Second Annual ACM Conference on Multimedia Systems*, MMSys '11, (New York, NY, USA), pp. 133–144, ACM, 2011.

[5] J. Zyren, "Overview of the 3GPP Long Term Evolution physical layer," Tech. Rep. 3GPPEVOLUTIONWP, Freescale Semiconductor, Inc, July 2007.

[6] 3GPP, TS 36.106, "FDD repeater radio transmission and reception," Release 8 v8.3.0, 2009-2010.

[7] A. Galanopoulos, T. Tsiftsis, and F. Foukalas, "Licensed Assisted Access: Key enabling functionalities and initial results," in *2015 International Symposium on Wireless Communication Systems (ISWCS)*, pp. 171–175, Aug 2015.

[8] A. Galanopoulos, G. Iosifidis, A. Argyriou, and L. Tassiulas, "Green video delivery in LTE-based heterogeneous cellular networks," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2015 IEEE 16th International Symposium on a*, pp. 1–9, June 2015.

[9] X. Wang, J. Chen, A. Dutta, and M. Chiang, "Adaptive video streaming over whitespace: SVC for 3-tiered spectrum sharing," in *Computer Communications (INFOCOM), 2015 IEEE Conference on*, pp. 28–36, April 2015.

[10] A. Abdel-Hadi and T. C. Clancy, "An optimal resource allocation with joint carrier aggregation in 4G-LTE," *CoRR*, vol. abs/1405.6448, 2014.

[11] T. Erpek, A. Abdel-Hadi, and T. C. Clancy, "An optimal application-aware resource block scheduling in LTE," *CoRR*, vol. abs/1405.7446, 2014.

[12] J. Chen, R. Mahindra, M. A. Khojastepour, S. Rangarajan, and M. Chiang, "A scheduling framework for adaptive video delivery over cellular networks," in *Proceedings of the 19th Annual International Conference on Mobile Computing & Networking*, MobiCom '13, (New York, NY, USA), pp. 389–400, ACM, 2013.

[13] S. Wang, Q. Yang, W. Shi, and C. Wang, "Interference mitigation and resource allocation in cognitive radio-enabled heterogeneous networks," in *Global Communications Conference (GLOBECOM), 2013 IEEE*, pp. 4560–4565, Dec 2013.

[14] D. Bethanabhotla, G. Caire, and M. J. Neely, "Utility optimal scheduling and admission control for adaptive video streaming in small cell networks," *CoRR*, vol. abs/1305.3586, 2013.

[15] D. Bethanabhotla, G. Caire, and M. J. Neely, "Joint transmission scheduling and congestion control for adaptive video streaming in small-cell networks," *CoRR*, vol. abs/1304.8083, 2013.

[16] D. Bethanabhotla, G. Caire, and M. J. Neely, "Adaptive video streaming for wireless networks with multiple users and helpers," *IEEE Transactions on Communications*, vol. 63, pp. 268–285, Jan 2015.

[17] Y. Li, F. Baccelli, J. G. Andrews, T. D. Novlan, and J. C. Zhang, "Modeling and analyzing the coexistence of Wi-Fi and LTE in unlicensed spectrum," *CoRR*, vol. abs/1510.01392, 2015.

[18] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 535–547, March 2000.

[19] G. Bianchi, "IEEE 802.11-saturation throughput analysis," *IEEE Communications Letters*, vol. 2, pp. 318–320, Dec 1998.

[20] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli, "A control-theoretic approach for dynamic adaptive video streaming over HTTP," *SIGCOMM Comput. Commun. Rev.*, vol. 45, pp. 325–338, Aug. 2015.

[21] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, pp. 1–122, Jan. 2011.

[22] R. Nishihara, L. Lessard, B. Recht, A. Packard, and M. I. Jordan, "A general analysis of the convergence of ADMM," *International Conference on Machine Learning*, 2015.

[23] M. Carpin, A. Zanella, J. Rasool, K. Mahmood, O. Grøndalen, and O. N. Østerbø, "Scheduling policies for the LTE downlink channel: A performance comparison," *CoRR*, vol. abs/1409.8633, 2014.

[24] M. J. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan & Claypool, 2010.

[25] M. T. Kawser, N. I. B. Hamid, M. N. Hasan, M. S. Alam, and M. M. Rahman, "Downlink SNR to CQI mapping for different multiple antenna techniques in LTE," *International Journal of Information and Electronics Engineering*, vol. 2, no. 5, 2012.