Joint Time-Domain Resource Partitioning, Rate Allocation, and Video Quality Adaptation in Heterogeneous Cellular Networks

Antonios Argyriou, *Member, IEEE*, Dimitrios Kosmanos, Leandros Tassiulas, *Fellow, IEEE*

***Abstract*—Heterogenous cellular networks (HCN) introduce small cells within the transmission range of a macrocell. For the efficient operation of HCNs it is essential that the high power macrocell shuts off its transmissions for an appropriate amount of time in order for the low power small cells to transmit. This is a mechanism that allows time-domain resource partitioning (TDRP) and is critical to be optimized for maximizing the throughput of the complete HCN. In this paper, we investigate video communication in HCNs when TDRP is employed. After defining a detailed system model for video streaming in such a HCN, we consider the problem of maximizing the experienced video quality at all the users, by jointly optimizing the TDRP for the HCN, the rate allocated to each specific user, and the selected video quality transmitted to a user. The NP-hard problem is solved with a primal-dual approximation algorithm that decomposes the problem into simpler subproblems, making them amenable to fast well-known solution algorithms. Consequently, the calculated solution can be enforced in the time scale of real-life video streaming sessions. This last observation motivates the enhancement of the proposed framework to support video delivery with dynamic adaptive streaming over HTTP (DASH). Our extensive simulation results demonstrate clearly the need for our holistic approach for improving the video quality and playback performance of the video streaming users in HCNs.**

***Index Terms*—Heterogeneous cellular networks, small cells, intra-cell interference, video streaming, video distribution, DASH, rate allocation, resource allocation, optimization, 5G wireless networks.**

I. Introduction

NEARLY 50% of the traffic in cellular networks today is video [1]. Mounting evidence suggests that video will keep increasing its share of the cellular traffic at an even faster pace [1]. The reason behind this phenomenon is the explosive demand for high quality video streaming from mobile devices (e.g., tablets, smart-phones). The challenge for mobile network operators (MNOs) is to offer higher data rates that can keep up with this demand for high quality video. Heterogenous cellular networks (HCNs), illustrated in Fig. 1, are envisioned to be one of the solutions to this problem. HCNs introduce low power base stations (BS) like pico BS (PBS) and femto BS (FBS), that form around them picocells and femtocells respectively. Lower transmission power from these small cells reduces the transmission range and allows improved spatial reuse. Hence, the first novel feature of HCNs is the *higher wireless capacity* they offer to the complete macrocell. The second novel feature of HCNs is that they can *lower the use of the backhaul capacity* by employing caching at the small cell BSs [2], [3]. Caching enables local access of frequently requested videos and this means lower utilization of the backhaul links between a BS and the video server. These two features of HCNs constitute them a central component of the envisioned 5G cellular network architecture.

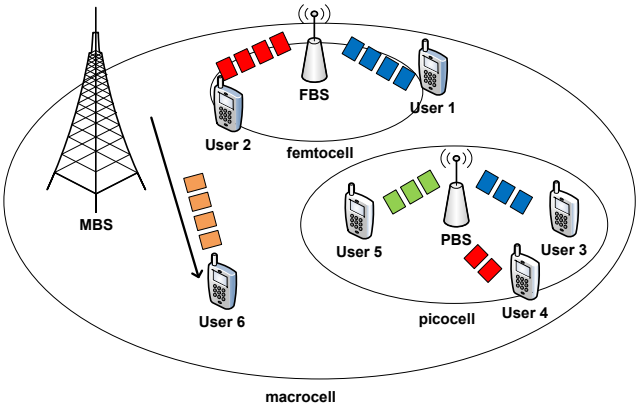

Fig. 1. The considered HCN consists of single macro and several pico and femto BSs. Each BS streams videos to a subset of the associated users.

Research for video distribution in HCNs has focused primarily on caching, with the objective to reduce the startup playback delay of the video for each user [2], [3], or lower the costs for the operator [3]. In this paper we are concerned with the first novel feature of HCNs which is the higher wireless capacity. We focus on this topic since HCNs introduce a new way for sharing the wireless resources. With the time domain resource partitioning (TDRP) mechanism [4], the MBS shuts off its transmissions for a fraction η of the available resources during which the small cells can achieve a higher data rate (Fig. 2). During the fraction $1-\eta$, there is *intra-cell* interference since the MBS transmits simultaneously with the small cells. This technique was recently standardized through the concept of almost blank subframes (ABS) and regular subframes (RS) in 3GPP LTE-A under the more general name of enhanced inter-cell interference coordination (eICIC) [4]. One important detail is that the LTE-A standard currently allows the dynamic adaptation of η but it does not specify how it should be configured. Given the increasing number of video streaming users in cellular networks, and the necessity of TDRP, it is of outmost importance to perform optimally both the configuration of η and the allocation of the wireless

Antonios Argyriou and D. Kosmanos are with the Department of Electrical and Computer Engineering, University of Thessaly, Greece. L. Tassiulas is with Yale University, USA.

resources in an HCN. Hence, the specific questions that should be answered in this case are: What is the optimal η when we have video traffic? What is the best video quality that each user should receive? What happens when a subset of the users receive video? Currently there are no definite answers to these pressing questions.

Related work. TDRP for HCNs is a topic investigated only recently because ABS/RS were also very recently standardized in LTE-A. The authors in [5] derived the optimal fraction from the available ABS and RS resources that each user should be allocated (a representative rate allocation is illustrated in Fig. 2) under a proportionally fair rate allocation (PFRA) metric. The authors of that work assumed a constant fraction of ABS η that is configured by the HCN operator. In another recent work reported in [6], the authors investigated the joint optimization of TDRP and user association (for traffic offloading) but with an assumption for equal rate allocation to the associated users. To the best of our knowledge there is no work that addresses TDRP in HCNs for video distribution and streaming. As we already discussed, much of the early research work for video streaming HCNs has focused on caching [2], [3], or exploiting particular features like the density of the small cells [7]. However, these works assumed the availability of a constant fraction of the resources for the MBS and the small cells that is effectively translated to a constant η. On the other hand, multi-user rate allocation for video streaming has been a topic thoroughly investigated the last few years for specific types of wireless networks. The works were primarily motivated from the network utility maximization (NUM) framework [8]. From the category of works that were based on NUM, the ones that are more related to this paper focused on cellular networks and considered more details of the physical layer (PHY). For example the authors in [9] investigated scheduling and resource allocation for a downlink LTE system that employs discrete decisions for optimizing the selected video streaming quality. The same problem, but for scalable encoded video, was considered in [10]. Another class of works focused on optimizing rate/resource allocation with the objective to improve the playback performance of video clients [11], [12]. In the last works the authors take into consideration recent standardization developments in streaming and in particular dynamic adaptive streaming over HTTP (DASH). However, these works do not target HCNs and assume access to fixed capacity resources.

Contributions. In this paper, we present contributions on three fronts. First, we present a comprehensive *Joint TDRP, Rate Allocation, and Video Quality Selection (JTRAVQS)* optimization framework for video streaming in HCNs. The framework identifies the optimal TDRP η, the rate allocated to each user, and the video quality description for each user, so as to maximize the aggregate video quality in the HCN. Our framework includes additional system-level parameters like the fraction of users that receive video. The NP-hard problem is solved with a primal-dual approximation algorithm that provides an asymptotically optimal solution. Our solution approach decomposes the problem into simpler subproblems, making them amenable to fast well-known solution algorithms. Second, we propose enhancements to the basic optimization framework that allow it to support DASH-based video streaming. Additional system parameters like the buffer contents of individual users, time-dependent user population, and channel capacity with TCP, are taken into account to optimize the the playback performance [13]. Third, we present a thorough performance evaluation our main framework against: a) A video-unaware system that jointly optimizes the TDRP and rate allocation under a PFRA metric [5]. b) From the results of our scheme and that of PFRA, we can infer the performance of a system that applies optimized rate allocation and video quality selection (RAVQS) but with a fixed TDRP [14]. Finally, our enhanced system for DASH, that considers the content of the playback buffer, is compared against a buffer-aware system that again uses fixed resources.

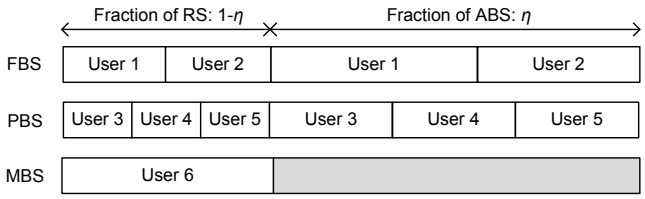

Fig. 2. Modeling the global time-domain resource partitioning, and local rate allocation problem in a topology with two PBSs and a MBS.

Main Results. Our results reveal that: i) For video streaming TDRP should be more aggressive in favor of the small cells when compared to TDRP optimization under a PFRA metric. In particular even for 4 small cells and 100 users, the optimal η should be nearly 22.2% higher than the optimal η under PFRA. Video quality is improved by a factor of 50%-70% for this scenario. ii) Using a fixed *but still optimal TDRP under a PFRA metric*, and then performing a RAVQS optimization as an afterthought, is still suboptimal. In particular for a population of 100 users and 4 small cells, the previous approach leads to an average video quality loss of 18.6% when compared to our approach. iii) For a DASH-based system with a fixed, *but again optimal TDRP under PFRA*, our optimization has more significant impact. In particular the rebuffering time/events of the clients can be reduced by more than 50% for a static network and 60% for a network with user churn.

Paper Organization. The rest of the paper is organized in the following sections. Section II describes in detail the system model. In Section III we present the formulation and the solution of the optimization problem we introduce in this paper, while its extension for DASH is presented in Section IV. Performance evaluation results are presented in Section V, and finally we conclude in Section VI.

II. System Model and Assumptions

Network Model. In Fig. 1 we present the network that we study in this paper and it includes a single macrocell with a MBS, the PBSs, and the users. Each BS j in the set $\mathcal{J}$ communicates with the set of associated users $\mathcal{N}_j$. We also denote with $\mathcal{F}_j \subseteq \mathcal{N}_j$ a subset of the users associated to BS j that are not optimized in a video-aware fashion. This parameter allows us to investigate the possibility that a fraction $f_j = (|\mathcal{N}_j| - |\mathcal{F}_j|)/|\mathcal{N}_j|$ of the users are optimized. During

the fraction η of the ABS resources all the small cells transmit and interfere with every active user in the network. Thus, we consider *resource reuse* across BSs of the same tier (PBSs in our case) which is one of the main benefits of small cells since it allows spatial reuse. The aggregate average interference power that user i receives is denoted as $I_{\text{ABS},i}$. During the fraction of the non-blanked resources, or regular subframes, $1-\eta$ both the MBS and PBSs transmit and the aggregate interference power that a user receives is denoted as $I_{\text{RS},i}$.

User Model. Each BS j transmits with unicast streaming video i to the similarly denoted user. The users associate to a BS by using an signal-to-interference plus noise ratio (SINR) biasing rule [6], i.e., a user is associated to the small cell j, and not the MBS, if the following is true: $SNR_{\text{PBS}_j} + Bias \geq SNR_{\text{MBS}}$. This ensures that users are offloaded to the small cells [6]. Our primary objective is a static user population similarly to the literature [2], [9], [11], [12], since we are interested to optimize the system operation within the complete playback duration of the video. However, motivated by recent experimental results that identify slow user variations in the cell throughout the day [15], we also evaluate our system for this more dynamic scenario.

Video Streaming and Playback Model. Without loosing generality we assume that all the BSs are assumed to have cached the videos for all the users [2], [3].[1] Now if the video representation that is transmitted to user i is indexed by r, the average bitrate that must be sustained is

$$R_{ir} = \frac{S_{ir}}{T_i + B_{i0}} \text{ bits/sec}, \tag{1}$$

where S_{ir} is the size of the r-th video representation, T_i is the total playback time of the video and B_{i0} is the duration of playable content received during startup buffering (time 0). This formulation ensures that the average probability of rebuffering events is zero [12]. In the first part of our optimization in Section III, this is the condition we adopt since we are interested to reach a decision once for the duration of the video streaming session. However, with DASH this formula is revised in Section IV.

Channel & PHY Model. Nodes use a single omni-directional half-duplex antenna. The channel from the j-th BS to the i-th user is denoted as $h_{j,i}$. The fading coefficients are independent and $h_{j,i} \sim \mathcal{CN}(PL_{i,j}, 1)$, i.e., they are complex Gaussian random variables with unit variance and mean equal to $PL_{i,j}$ that depends on path loss and shadowing effects according to the LTE channel model [4]. All the channels are considered to be block-fading Rayleigh and quasi-stationary, that is they remain constant for the coherence period of the channel that is equal to the transmission length of the complete PHY block. Additive white Gaussian noise (AWGN) is assumed at every receiver with variance σ^2. The transmission power that the PBS and MBS use is P_{PBS}, and P_{MBS} respectively.

MCS & CQI. A modulation and coding scheme (MCS) with m bits/symbol is used by each BS while its value is determined by each PBS independently and optimally as we will later explain. The set of available MCSs is $\mathcal{M} = \{1, ..., 7\}$, i.e., we assume that the most spectral efficient quadrature amplitude modulation (QAM) MCS is 128-QAM. We also assume that users provide only *average channel quality indicator* (CQI) feedback to the BSs.

A. Video Quality Model

Modeling the Quality-of-Experience (QoE) of users in video streaming applications is not easy. QoE is affected both by the video signal quality and delay [16], [13]. In this subsection, we define a utility model only for the quality of the video signal while during the analysis of our optimization framework we discuss our approach for minimizing the effects of delay. The main objective of our video quality model is to capture the rate-distortion (RD) relationship of different representations of each video stream. This will allow our optimization framework to allocate resources to videos depending on their quality.

In this paper we assume we have the RD information information for each frame n that belongs to representation r of video i and consists of its size S_{irn} in bits and the importance of the frame for the overall reconstruction quality of the video denoted as q_{irn} [17]. In practice, q_{irn} is the total decrease in the mean square error (MSE) distortion that will affect the video if the frame is decoded by the video player [18]. The value of the MSE in q_{irn} includes both the distortion that is added when frame n is not decoded, and also the frames that have a decoding dependency with n.[2] Hence, the video quality model considers also the possible drift that might occur due to the inability to decode a particular video frame. These values can be obtained easily but only during the offline encoding of the video as discussed in [19], [20].

Consequently, the aggregate video quality of a group of video frames indexed by s (also referred to as segment to ensure consistency with DASH terminology), that belong to representation r of video i, is the average MSE reduction/frame:

$$q_{irs} = \frac{\sum_n q_{irn}}{\text{number of frames}} \tag{2}$$

This fraction is the average MSE reduction of the frames contained in a DASH segment or packet, versus their total number. *This formulation is in line with our initial objective since it expresses the "value" for a group of frames.* For a group of segments starting from t-th segment until the end of the video, we can similarly characterize the video quality as:

$$Q_{irt} = \frac{\sum_{s=t}^{s=\text{last}} q_{irs}}{\text{number of remaining segments}} \tag{3}$$

For packet-based video, this RD information associated with a packet can be contained in each packet header. In the case of scalable video the information about the importance of a packet is already embedded in the header since it indicates the video layer that the packet belongs. For segment-based DASH streaming a media presentation description (MPD) file is already used for conveying a subset of this information [21].

[1] Our system can easily accommodate the case that the video stream originates from a server by considering end-to-end throughput that the network can deliver.

[2] For example q for an I frame includes the q of the P and B frames that depend on it.

Hence, the model can support packetized non-scalable, scalable, and segment-based video. The final result of the previous discussion is that a single video for user i will be available at the following discrete set of qualities

$$\mathcal{Q}_{it} = \{Q_{i1t}, .., Q_{irt}, ..., \} \text{ with } r \in \mathcal{R}_i \quad (4)$$

indicating the set of available representations for each user/video i. It is important to understand the use of the previous model in our optimization. In our initial framework, where the problem is solved for the complete playback duration of the video, the formulation in (3) is used by setting t=0, i.e., we use the average quality of the complete video. However, for DASH the optimization is solved during a specific time period t and (3) captures the video quality of the remaining segments that still need to be communicated.

To complete our discussion, we have to recall that our optimization targets a heterogeneous user population where a subset of them do not receive video. When we have elastic flows, or when the users do not participate in the video-aware optimization, then rate allocation is exercised with a PFRA metric [6], i.e., Q_{irt} is generated by taking the logarithm of the communication rate achievable by user i.

B. Throughput at the Physical Layer

We consider that the BSs optimize independently the PHY parameters of the point-to-point links, as it is typically done in wireless communication systems [22]. To estimate the average communication rate that each user i achieves when it is associated to BS j we proceed as follows. The BS receives from each user i the average channel gain $\mathbb{E}[|h_{j,i}|^2]$, and also the local estimate of the interference power $I_{\text{ABS},i}$, $I_{\text{RS},i}$.[3] Note that the average channel gains, or the CQI in the LTE terminology, mentioned above can be transmitted from each user to the BS with low network overhead since they only correspond to path loss and shadowing. Hence, during an ABS, since a user receives the aggregate interference from all the simultaneously transmitting PBSs, the average SINR between the PBS and user i is:

$$\mathbb{E}[\gamma_i^{\text{ABS}}] = \frac{P_{\text{PBS}}\,\mathbb{E}[|h_{\text{PBS},i}|^2]}{I_{\text{ABS},i} + \sigma^2} \quad (5)$$

During the RS the MBS is also active and so the SINR of users associated to a PBS and the MBS are

$$\mathbb{E}[\gamma_i^{\text{RS}}] = \frac{P_{\text{PBS}}\,\mathbb{E}[|h_{\text{PBS},i}|^2]}{I_{\text{RS},i} + \sigma^2}, \text{ and } \mathbb{E}[\gamma_i^{\text{RS}}] = \frac{P_{\text{MBS}}\,\mathbb{E}[|h_{\text{MBS},i}|^2]}{I_{\text{RS},i} + \sigma^2},$$

respectively. The average SINR expressions allow each BS j to estimate the resulting average data rate for each associated user i under MCS m as:

$$C_{im} = m \cdot eff \cdot S \cdot (1 - P_s)^{L/m} \text{ bits/sec}, \quad (6)$$

where S is the symbol rate, eff is the efficiency of the MCS, and the probability of symbol error P_s under 2^m-QAM is [23]:

$$P_s = 4(1 - 2^{m/2})Q\Big(\sqrt{\frac{3}{2^m - 1}\,\mathbb{E}[\gamma_i]}\Big) \quad (7)$$

In our system the PHY and link-layer system at each BS selects optimally for the average SINR the MCS that ensures the highest point to point communication rate [24]. This is formally written as:

$$C_i = \max_{m \in \mathcal{M}} C_{im} \quad (8)$$

[3]LTE Rel. 8 already implements the communication of the power of the local interference though the high interference indicator (HII).

III. Problem Formulation and Solution

Now we are ready to define formally the problem we address in this paper. For each user i associated to BS j the HCN must select the video representation with the highest quality, and the rate allocated to it. For the complete HCN the globally optimal TDRP must be calculated. First we define the optimization variables. Let $x_{ir}^{\text{ABS}}, x_{ir}^{\text{RS}} \in \{0, 1\}$ indicate whether user $i \in \mathcal{N}_j \cup \mathcal{F}_j$ is served with video representation r in an ABS and RS respectively. Let also $z_{ir}^{\text{ABS}} \in [0, 1]$ denote the fraction of the ABS resources that the PBS allocates to $i \in \mathcal{N}_j \cup \mathcal{F}_j$ for streaming the video representation r. Similarly for the RS, we define $z_{ir}^{\text{RS}} \in [0, 1]$. Hence, the decisions of each BS j are: (a) the *video quality selection (VQS)* vector for all the associated users, i.e., $\boldsymbol{x}_j = \big(x_{ir}^{\text{ABS}} \geq 0 : i \in \mathcal{N}_j \cup \mathcal{F}_j, r \in \mathcal{R}_i\big)$, and (b) the *rate allocation (RA)* vector for all users, i.e., $\boldsymbol{z}_j^{\text{ABS}} = \big(z_{ir}^{\text{ABS}} \geq 0 : i \in \mathcal{N}_j \cup \mathcal{F}_j, r \in \mathcal{R}_i\big)$. Similarly the VQS and RA vectors for the regular slots. Also the global resource partitioning decision η. To minimize the notation later in our solution, we also define different concatenations of the variable vectors as follows: $\boldsymbol{z}_j = \big(\boldsymbol{z}_j^{\text{ABS}}, \boldsymbol{z}_j^{\text{RS}}\big)$, $\boldsymbol{z} = \big(\boldsymbol{z}_j : j \in \mathcal{J}\big)$, and similarly for $\boldsymbol{x}_j, \boldsymbol{x}$.

The objective for the HCN operator is to maximize the average aggregate delivered video quality captured by:

$$\sum_{j \in \mathcal{J} \setminus \{0\}} \sum_{i \in \mathcal{N}_j \cup \mathcal{F}_j} \sum_{r \in \mathcal{R}_i} (x_{ir}^{\text{ABS}} + x_{ir}^{\text{RS}}) Q_{ir0} + \sum_{i \in \mathcal{N}_0 \cup \mathcal{F}_0} \sum_{r \in \mathcal{R}_i} x_{ir}^{\text{RS}} Q_{ir0} \quad (9)$$

In the above recall that Q_{ir0} is the average quality of representation r for video i. Thus, the objective expresses the video quality delivered to the complete HCN. In the second term we have the quality for the users associated to the MBS since they cannot transmit during an ABS.

For the first set of constraints we have to recall that the fraction of the blank ABS resources available for the PBSs (there is resource re-use across the PBSs) is η. This leads to:

$$\sum_{i \in \mathcal{N}_j \cup \mathcal{F}_j} \sum_{r \in \mathcal{R}_i} z_{ir}^{\text{ABS}} \leq \eta, \forall j \in \mathcal{J} \setminus \{0\} \quad (10)$$

In the above we excluded again the MBS since it cannot transmit during an ABS. During the RS all the BSs transmit:

$$\sum_{i \in \mathcal{N}_j \cup \mathcal{F}_j} \sum_{r \in \mathcal{R}_i} z_{ir}^{\text{RS}} \leq 1 - \eta, \forall j \in \mathcal{J} \quad (11)$$

When a particular representation r is selected, the average rate R_{ir} in bits/sec that must be sustained by a user i is less than the rate that can be achieved during both the ABS and RS. Also the resources allocated during ABS and RS will determine the average rate. The above can be formally written as:

$$x_{ir}^{\text{ABS}} R_{ir} \leq (z_{ir}^{\text{ABS}} C_i^{\text{ABS}} + z_{ir}^{\text{RS}} C_i^{\text{RS}}), \forall r \in \mathcal{R}_i, i \in \mathcal{N}_j \cup \mathcal{F}_j, j \in \mathcal{J} \quad (12)$$

In this section this constraint is based in (1), and in this form it ensures that the average number of rebuffering time over the complete video playback is zero. Also, (12) accounts for the startup delay as specified in (1). We will delve into the extension for DASH in the next section.

We also have that resources cannot be allocated to a video representation r if it is not actually selected:

$$z_{ir}^{\text{ABS}} \leq x_{ir}^{\text{ABS}}, \forall r \in \mathcal{R}_i, i \in \mathcal{N}_j \cup \mathcal{F}_j, j \in \mathcal{J} \quad (13)$$

$$z_{ir}^{\text{RS}} \leq x_{ir}^{\text{RS}}, \forall r \in \mathcal{R}_i, \forall i \in \mathcal{N}_j \cup \mathcal{F}_j, j \in \mathcal{J} \quad (14)$$

We also need the integer constraints according to which only one video representation r can be used for each user. Thus:

$$\sum_{r \in \mathcal{R}_i} x_{ir}^{\text{ABS}} \leq 1, \forall i \in \mathcal{N}_j \cup \mathcal{F}_j, j \in \mathcal{J} \quad (15)$$

$$\sum_{r \in \mathcal{R}_i} x_{ir}^{\text{RS}} \leq 1, \forall i \in \mathcal{N}_j \cup \mathcal{F}_j, j \in \mathcal{J} \quad (16)$$

During the regular slots the PBSs can also transmit together with the MBS, albeit with lower spectral efficiency. In this case the rate will be lower. However, we must ensure that across ABS and RS the same video representation is used:

$$x_{ir}^{\text{ABS}} = x_{ir}^{\text{RS}}, \forall r \in \mathcal{R}_i, \forall i \in \mathcal{N}_j \cup \mathcal{F}_j, j \in \mathcal{J} \quad (17)$$

The last condition comes from the observation that it is not practical to transmit one video quality during ABS and a different during the RS, since these two types of resources alternate at the PHY in the order of milliseconds [4].

A. Hierarchical Primal and Dual Decomposition

This problem formulation clearly constitutes a non-convex mixed integer linear program (MILP). Hence, it is NP-hard while it does not map to a well-known structure that can be solved with fast pseudo-polynomial algorithms (e.g., knapsack forms). This last aspect can also be seen fairly easily since we have that η, which is the capacity of the knapsack in (10),(11), is also an optimization variable. The second issue is the evident need for distributed computation. These aspects make the problem computationally challenging. Despite this difficult challenge at first sight, we notice that at this stage of our work we are interested to calculate the average rate allocation and TDRP during the complete streaming session. This means that in practice the final algorithm does not have to provide a solution in the order of seconds, but minutes. For this reason, instead of designing heuristics, we resort to a solution approach with a primal-dual approximation algorithm that converges asymptotically to the optimal solution [25].

To solve this problem we apply first primal decomposition on η. Primal decomposition consists of setting a constant value to the coupling variable [26]. For a constant value for η we notice that the original problem is decomposed into a master problem P_0, and several problems denoted as P_j (each one for each BS j). We use Lagrangian relaxation to solve P_j. The Lagrangian after relaxing the coupling constraints for P_j is:

$$\begin{aligned} L_j = & \sum_{i \in \mathcal{N}_j \cup \mathcal{F}_j} \sum_{r \in \mathcal{R}_i} (x_{ir}^{\text{ABS}} + x_{ir}^{\text{RS}}) Q_{ir0} + \boldsymbol{\lambda}_{1,j}^{\text{ABS}} \boldsymbol{f}_1^{\text{ABS}}(\boldsymbol{z}_j^{\text{ABS}}) \\ & + \boldsymbol{\lambda}_{1,j}^{\text{RS}} \boldsymbol{f}_1^{\text{RS}}(\boldsymbol{z}_j^{\text{RS}}) + \boldsymbol{\eta}(\boldsymbol{\lambda}_{1,j}^{\text{RS}} - \boldsymbol{\lambda}_{1,j}^{\text{ABS}}) - \boldsymbol{\lambda}_{1,j}^{\text{RS}} \mathbf{1} + \boldsymbol{\lambda}_{2,j} \boldsymbol{f}_2(\boldsymbol{z}_j^{\text{ABS}}) \\ & + \boldsymbol{\lambda}_{2,j} \boldsymbol{f}_2(\boldsymbol{z}_j^{\text{RS}}) + \boldsymbol{\lambda}_{2,j} \boldsymbol{f}_2(\boldsymbol{x}_j^{\text{ABS}}) + \boldsymbol{\lambda}_{3,j}^{\text{ABS}} \boldsymbol{f}_3^{\text{ABS}}(\boldsymbol{z}_j^{\text{ABS}}) \\ & + \boldsymbol{\lambda}_{3,j}^{\text{ABS}} \boldsymbol{f}_3^{\text{ABS}}(\boldsymbol{x}_j^{\text{ABS}} + \boldsymbol{\lambda}_{3,j}^{\text{RS}} \boldsymbol{f}_3^{\text{RS}}(\boldsymbol{z}_j^{\text{RS}}) + \boldsymbol{\lambda}_{3,j}^{\text{RS}} \boldsymbol{f}_3^{\text{RS}}(\boldsymbol{x}_j^{\text{RS}}) \\ & + \boldsymbol{\lambda}_{4,j}^{\text{ABS}} \boldsymbol{f}_4^{\text{ABS}}(\boldsymbol{x}_j^{\text{ABS}}) + \boldsymbol{\lambda}_{4,j}^{\text{RS}} \boldsymbol{f}_4^{\text{RS}}(\boldsymbol{x}_j^{\text{RS}}) \\ & + \boldsymbol{\lambda}_{5,j} \boldsymbol{f}_5(\boldsymbol{x}_j^{\text{ABS}}) + \boldsymbol{\lambda}_{5,j} \boldsymbol{f}_5(\boldsymbol{x}_j^{\text{RS}}) \end{aligned} \quad (18)$$

In the above $\boldsymbol{f}_1^{\text{ABS}}, \boldsymbol{f}_1^{\text{RS}}$ are constraint vectors (10), (11), constraint (12) is expressed with vector $\boldsymbol{f}_2$, and (13),(14) are written as the constraint vectors $\boldsymbol{f}_3^{\text{ABS}}$,$\boldsymbol{f}_3^{\text{RS}}$, while $\boldsymbol{f}_4^{\text{ABS}}$,$\boldsymbol{f}_4^{\text{RS}}$ correspond to (15),(16). Finally $\boldsymbol{f}_5$ corresponds to (17). The dual variables $\boldsymbol{\lambda}$ are all in row vector form in order to avoid the need for a transpose superscript. Now by using this relaxation, and by packing all the Lagrangian multipliers for PBS j as the single vector $\boldsymbol{\lambda}_j$, the dual problem can be written as follows:

$$\min_{\boldsymbol{\lambda}_j} \max_{\boldsymbol{z}_j, \boldsymbol{x}_j} L_j(\boldsymbol{z}_j, \boldsymbol{x}_j, \boldsymbol{\lambda}_j) \text{ s.t. } (10) - (17), \boldsymbol{\lambda}_j \geq 0 \quad (19)$$

A constant η after primal decomposition has further implications. In particular the problem in (19) is decomposed into subproblems for the ABS and RS. We also notice from L_j that these subproblems can be further decomposed for both the ABS and the RS. First we have a RA problem $\text{P}_j^{\text{ABS-RA}}$:

$$\begin{aligned} & \max_{\boldsymbol{z}_j^{\text{ABS}}} (\boldsymbol{\lambda}_{1,j}^{\text{ABS}} \boldsymbol{f}_1^{\text{ABS}}(\boldsymbol{z}_j^{\text{ABS}}) - \boldsymbol{\eta} \boldsymbol{\lambda}_{\mathbf{1},\boldsymbol{j}}^{\text{ABS}} + \boldsymbol{\lambda}_{2,j} \boldsymbol{f}_2(\boldsymbol{z}_j^{\text{ABS}}) \\ & + \boldsymbol{\lambda}_{3,j}^{\text{ABS}} \boldsymbol{f}_3^{\text{ABS}}(\boldsymbol{z}_j^{\text{ABS}})) \text{ s.t. } (10), (12), (13) \end{aligned} \quad (20)$$

Also a VQS problem, denoted as $\text{P}_j^{\text{ABS-VQS}}$:

$$\begin{aligned} & \max_{\boldsymbol{x}_j^{\text{ABS}}} (\sum_{i \in \mathcal{N}_j \cup \mathcal{F}_j} \sum_{r \in \mathcal{R}_i} x_{ir}^{\text{ABS}} Q_{ir0} + \boldsymbol{\lambda}_{2,j} \boldsymbol{f}_2(\boldsymbol{x}_j^{\text{ABS}}) + \boldsymbol{\lambda}_{3,j} \boldsymbol{f}_{\mathbf{3}}(\boldsymbol{x}_j^{\text{ABS}}) \\ & + \boldsymbol{\lambda}_{4,j} \boldsymbol{f}_{\mathbf{4}}(\boldsymbol{x}_j^{\text{ABS}}) + \boldsymbol{\lambda}_{5,j} \boldsymbol{f}_{\mathbf{5}}(\boldsymbol{x}_j^{\text{ABS}})) \text{ s.t. } (12), (13), (15), (17) \end{aligned} \quad (21)$$

Thus, we have two linear RA and two integer VQS problems that are solved by each BS as we explain next.

B. Rate Allocation and Video Quality Selection at the BSs

The dual problem is solved in an iterative fashion, using a primal-dual Lagrange method that can allow us to reach an asymptotically optimal solution [25], [3]. The central concept of the primal-dual algorithm is to initialize first the dual variables to zero, and then to solve subproblems $\text{P}_j^{\text{ABS-RA}}$, $\text{P}_j^{\text{ABS-VQS}}$, $\text{P}_j^{\text{RS-RA}}$, $\text{P}_j^{\text{RS-VQS}}$ to obtain the currently optimal solution for iteration τ as $\boldsymbol{z}_j(\tau)$, and $\boldsymbol{x}_j(\tau)$. Besides our problem-specific details described in this subsection, further details regarding the primal-dual method and its asymptotically optimal convergence property can be found in [25], [3].

First we focus on subproblem (20) that is linear program. Hence, it can be efficiently solved using standard convex optimization techniques [25]. The BS also solves the second subproblem, i.e., the integer linear program (ILP) in (21) for identifying the currently optimal video representation for each user. This problem is solved in pseudo-polynomial time using dynamic programming (DP) [25]. Its speed of convergence is evaluated collectively for the complete system in our performance evaluation, while the time complexity of the complete JTRAVQS is discussed in a later subsection.

After solving the subproblems, and given the current result $\boldsymbol{z}_j^{\text{ABS}}(\tau), \boldsymbol{x}_j^{\text{ABS}}(\tau)$, we employ a sub-gradient method [25] to update the dual variables. A representative calculation is presented for the dual variables of the very first constraint:

$$\lambda_{1,jir}(\tau + 1) = \left[\lambda_{1,jir}(\tau) + \beta(\tau)(z_{ir}^{\text{ABS}}(\tau) - \eta) \right]^+ \quad (22)$$

In the above, the term in the parenthesis is the sub-gradient, $[.]^+$ denotes the projection onto the non-negative orthant, and

$\beta(\tau)$ is the step size at iteration τ. Similarly we define the update rules and the subgradients for the remaining dual variables. In each iteration τ, the dual objective is improved using the subgradient update and accordingly the primal relaxed problems $\mathrm{P}_j^{\text{ABS-RA}}, \mathrm{P}_j^{\text{ABS-VQS}}, \mathrm{P}_j^{\text{RS-RA}}, \mathrm{P}_j^{\text{RS-VQS}}$ are solved again in order to update the primal variables (which are then used in the subsequent dual objective update).

C. Solving the Master Problem

For the final step, the dual variables for constraints of each P_j are provided to the master problem P_0 that has to be solved now for the optimal η at the central controller (CC) of the system. Recall that for the master problem we applied primal decomposition. It is thus solved very efficiently by collecting only the resource prices for η, i.e $\lambda_{1,j}^{\text{ABS}}(\tau), \lambda_{1,j}^{\text{RS}}(\tau)$, from each PBS in order to form the global subgradient [26]. In practice we only transmit the local subgradient $\lambda_{1,j}^{\text{ABS}}(\tau) - \lambda_{1,j}^{\text{RS}}(\tau)$ from each BS. The current value of η is updated as follows [26]:

$$\eta(\tau+1) = \Big[\eta(\tau) + \boldsymbol{\beta}(\tau) \underbrace{\begin{bmatrix} 0 - \lambda_{1,0}^{\text{RS}}(\tau) \\ \cdots \\ \lambda_{1,j}^{\text{ABS}}(\tau) - \lambda_{1,j}^{\text{RS}}(\tau) \\ \cdots \end{bmatrix}}_{\text{global subgradient}} \Big]^+ \tag{23}$$

Now $\boldsymbol{\beta}(\tau)$ is a vector that can be selected as before in order to control the speed of convergence [25].

D. Discussion on Time Complexity

The time complexity of the discrete knapsack problem denoted by $\mathrm{P}_j^{\text{ABS-VQS}}$, when solved with dynamic programming, is polynomial with respect to the size of the problem $|\mathcal{N}_j||\mathcal{R}|$ for each BS j, but for bounded knapsack capacity. Hence, in our case it is $\mathcal{O}(|\mathcal{N}_j| \cdot |\mathcal{R}| \cdot 1)$, i.e., linear. This is because from (15) we notice that the capacity of the knapsack is 1, in other words only one item (video representation) can be selected. $\mathrm{P}_j^{\text{ABS-RA}}$ is a LP and so polynomial with respect to the number of associated users $|\mathcal{N}_j|$, i.e., $\mathrm{P}_j^{\text{ABS-RA}}$ is $\mathcal{P}(|\mathcal{N}_j|)$. Hence, the time complexity of $\mathrm{P}_j^{\text{ABS-VQS}}$ may be better than that of $\mathrm{P}_j^{\text{ABS-RA}}$. We conclude that the worst execution time of one iteration of the JTRAVQS algorithm, as a function of its inputs, can be expressed as:[4]

$$\max_{j\in\mathcal{J}} \Big(\max\big(\mathcal{O}(|\mathcal{N}_j||\mathcal{R}|), \mathcal{P}(|\mathcal{N}_j|)\big) + d_{j,CC} \Big) + \mathcal{O}(1) + \max_{j\in\mathcal{J}}(d_{CC,j}),$$

where $d_{j,CC}$ is the delay between BS j and the CC, $\mathcal{O}(1)$ corresponds to the execution time of the primal update (one vector multiplication and one addition) and $d_{CC,j}$ is the delay for communicating the new primal update to the BSs. In our simulation we also considered that the backhaul links have the *worst case* transmission delay of $d_{j,CC}$=60ms [27]. For 100 iterations the total delay until the optimal solution is reached is 6 seconds which means that the calculations have to take place within 4 seconds in order to reach a solution within a 10 second period. This is well within the capabilities of modern processors for solving the discussed LP and ILP. Also, each BS j communicates only $\lambda_{1,j}^{\text{ABS}}, \lambda_{1,j}^{\text{RS}}$ to the CC which constitutes a negligible overhead. The good performance of the algorithm, allow us to investigate its use in shorter time scales next.

[4]The quantities in the $\mathcal{O}, \mathcal{P}$ notation have to be multipled with the execution time of the fundamental algorithm operation.

IV. Problem Formulation and Solution for DASH

Motivation. In a network it is possible that channel conditions and users are more dynamic. In this case the bitrate of the transmitted video should be adapted. One way to accomplish that is DASH. With DASH a video is stored as a sequence of short duration (typically 2-10 sec) video segments [28]. Each segment may be available at different sizes, SNR qualities, spatial resolutions, frame rates. However, it has been shown that allowing the client to be fully responsible for requesting video segments (a pull-based system), after estimating the variations in the end-to-end throughput, results in significant waste of resources [29], [30]. In this paper, our optimization framework at the BS is responsible for the choice of the optimal video representation. This is also a realistic option since DASH does not specify where video adaptation occurs.

Enhanced System Model. We define the term *slot* as the period that the problem is solved and its decisions are enforced (see Fig. 3). Without loosing generality we assume that JTRAVQS-DASH is solved during a slot with a duration of 10 seconds. Since the problem is solved for every slot, the instance of the problem currently solved is also indexed by t. The result is that the JTRAVQS-DASH problem is solved during slot t to calculate the optimal RA and VQS for slot $t+1$. The algorithm requires several iterations as before, that are indexed also by τ. Regarding the input parameters to JTRAVQS-DASH C_{it} is our estimate of the TCP throughput of user i during slot t according to [31], and $\mathcal{N}_{jt}$ the set of associated users. This approach for modeling C_{it}, is consistent with the behavior of DASH that uses TCP for downloading segments. Hence, contrary to related work our approach is more realistic with respect to C_{it} [12]. The video quality model in (3) is used with Q_{irt} denoting the quality of the remaining segments that should be transmitted. Let us finally define some minimal additional notation since the optimization variables must be indexed by slot t: $\boldsymbol{z}_{jt} = \big(z_{irt}^{\text{ABS}}, z_{irt}^{\text{RS}} \geq 0 : i \in \mathcal{N}_{jt} \cup \mathcal{F}_{jt}, r \in \mathcal{R}_i\big)$ and $\boldsymbol{x}_{jt} = \big(x_{irt}^{\text{ABS}} \geq 0 : i \in \mathcal{N}_{jt} \cup \mathcal{F}_{jt}, r \in \mathcal{R}_i\big)$. Now x_{irt}^{ABS} indicates that in slot t segments from representation r will be transmitted to user i. The same algorithm is used for solving JTRAVQS-DASH but the two subproblems are adapted.

DASH Rate Allocation (DASHRA) Problem. The most important aspect is the re-formulation of constraint (12) that is now indexed by the slot t, and is packed into constraint vector $\boldsymbol{f}_2^{\text{DASH}}$:

$$x_{irt}^{\text{ABS}} S_{irt} \leq (z_{irt}^{\text{ABS}} C_{it}^{\text{ABS}} + z_{irt}^{\text{RS}} C_{it}^{\text{RS}}) \max\{\Delta B_{it}, 1\}, \forall r \in \mathcal{R}_i, i \in \mathcal{N}_{jt} \tag{24}$$

Re-writing (20) for the DASH case yields the problem $\mathrm{P}_{jt}^{\text{ABS-DASHRA}}$:

$$\max_{\boldsymbol{z}_{jt}^{\text{ABS}}} (\boldsymbol{\lambda}_{1,j}^{\text{ABS}} \boldsymbol{f}_1^{\text{ABS}}(\boldsymbol{z}_{jt}^{\text{ABS}}) - \boldsymbol{\eta}\boldsymbol{\lambda}_{1,j}^{\text{ABS}} + \boldsymbol{\lambda}_{2,j} \boldsymbol{f}_2^{\text{DASH}}(\boldsymbol{z}_{jt}^{\text{ABS}}, \boldsymbol{\Delta B}_t) + \boldsymbol{\lambda}_{3,j}^{\text{ABS}} \boldsymbol{f}_3^{\text{ABS}}(\boldsymbol{z}_{jt}^{\text{ABS}})) \text{ s.t. } (10), (24), (13) \tag{25}$$

In (24) S_{irt} is the average bitrate that must be sustained by the remaining segments of the r-th representation similarly to (1), (3). The key difference from the initial problem is parameter ΔB_{it}. This is the total duration of the playable video in seconds that user i has in its buffer at the start of slot t and is denoted as B_{it}, minus the playable video that the slowest user has in its buffer, i.e., $\Delta B_{it} = B_{it} - B_{(\text{slowest})t}$.

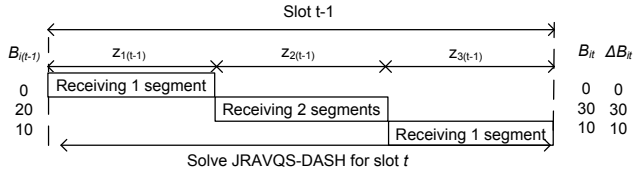

Fig. 3. Transmission of different segments during slot $t-1$ that has a durationof 10 seconds. JTRAVQS-DASH is solved to reach the decisions for the next slot t.

Each user updates the estimate of the playable video as: $B_{it} = B_{i(t-1)} +$ received during (t-1) $-$ played during (t-1). This parameter ensures that users that have received lower volume of data are effectively prioritized. Hence, if the RA decision for user i in the $t-1$-th slot is $z_{i(t-1)}$, then at the start of slot $t-1$ the BS can calculate B_{it}, since it knows the result of RA and of course the duration of video that will be played. To summarize, this is effectively a rebuffering constraint that contains the differential buffer information.

DASH Client Model Example. To explain the playback model for the DASH client and the setting of ΔB_{it}, let us use a specific example with clients that have different playback buffer contents as illustrated in Fig. 3. The number of transmitted segments depend on the value of $z_{i(t-1)}$. Also in this example we consider the downloading of complete segments for exposition purposes but our model supports partially downloaded segments. For user 1 assume that it is the client that is lagging behind from the rest and the playble video it has in its buffer at the start of slot $t-1$ is $B_{1(t-1)}$=0. Hence, at the start of slot $t-1$ it will rebuffer until it receives the segment and after it finishes, the video player enters the playback mode. Also B_{1t}=$B_{1(t-1)}$+10-10=0, and ΔB_{1t}=$B_{1t} - B_{1t}$=0. At the start of slot $t-1$ user 2 has 20 seconds worth of video, while during $t-1$ it will receive two additional segments leading to B_{2t}=20+20-10=30, and ΔB_{2t}=$B_{2t} - B_{1t}$=30. Hence, by having allocated more resources with $z_{2(t-1)}$ the result is a pre-fetching of data. For user 3 similarly we obtain B_{3t}=10+10-10=10 and $\Delta B_{2t} = B_{3t} - B_{1t}$=10.

DASH Video Quality Selection (DASHVQS) Problem. Now the VQS problem is solved by adding one constraint in the problem (21). We reformulate the $\mathrm{P}_j^{\text{ABS-VQS}}$ problem to $\mathrm{P}_{jt}^{\text{ABS-DASHVQS}}$ that is also solved over the $t-1$ slot:

$$\max_{\boldsymbol{x}_{jt}^{\text{ABS}}}\Big(\sum_{i\in\mathcal{N}_j\cup\mathcal{F}_j}\sum_{r\in\mathcal{R}_i} x_{irt}^{\text{ABS}}Q_{irt} + \boldsymbol{\lambda}_{2,j}\boldsymbol{f}_{\mathbf{2}}^{\text{DASH}}(\boldsymbol{x}_{jt}^{\text{ABS}}) + \boldsymbol{\lambda}_{3,j}\boldsymbol{f}_{\mathbf{3}}(\boldsymbol{x}_{jt}^{\text{ABS}}) + \boldsymbol{\lambda}_{4,j}\boldsymbol{f}_{\mathbf{4}}(\boldsymbol{x}_{jt}^{\text{ABS}}) + \boldsymbol{\lambda}_{5,j}\boldsymbol{f}_{\mathbf{5}}(\boldsymbol{x}_{j}^{\text{ABS}})\Big) \quad \text{s.t. (12), (13), (15), (17)}$$

Note that for partial downloading, if in the next slot DASHVQS identifies that a lower or higher video quality is transmitted, then pre-buffered data are not discarded but the unfinished segment is received. The new decision for the video quality is enforced when a new segment will be transmitted. The dual variables are updated as before and the sub-gradients are similarly modified based on the new constraint.

V. Performance Evaluation

In this section, we present a comprehensive evaluation of the proposed algorithms comprising our framework through custom Matlab simulation. Our simulator performs a precise PHY-level simulation of wireless packet transmissions.

JTRAVQS Evaluation. The parameter settings for our simulations are set as follows. Downlink MBS and PBS transmit power are equal to 46dBm and 30dBm respectively [6]. Distance-dependent path loss is given by $L(d) = 128.1 + 37.6\log_{10}(d)$, where d is the distance between two nodes in Km [4], and the shadowing standard deviation is 8 dB. The user speed is 3 kmph (quasi-static as we already stated), and average CQI is provided every 10 minutes. The macrocell area is set to be a circle with radius equal to 1 Km. The wireless channel parameters include a channel bandwidth of W=20 MHz, noise power spectral density of σ^2=10^{-6} Watt/Hz, while the same Rayleigh fading model was used for all the channels. Packets of 1500 bytes are transmitted at the PHY, while the optimal MCS is calculated according to (8). The user distribution and picocell locations are random and uniform within the macrocell. We set the biasing threshold to 0 dB for all the systems to calculate $\mathcal{N}_j$. The user population increases up to a number of 200 to evaluate the performance in networks that continuously become more dense, consistently with the recent trends [6]. For the PFRA system we configured the users to request randomly and uniformly one of the available video representations, while for JTRAVQS users request randomly and uniformly one of the available videos. The video content used in the experiments consists of 26 CIF (352x288), and high definition (1920x1080) sequences that were encoded with SVC H.264 as a single layers [20]. The videos are compressed at 30 fps and different rates ranging from 128 Kbps and reaching values$<$7 Mbps. The frame-type patterns were G16B1,G16B3,G16B7,G16B15, i.e., there are different numbers of B frames between every two P frames and a GOP size is always equal to 16 frames.

Regarding the presentation of the results, Fig. 4 shows the average video quality (in terms of the representation r) that is delivered to the picocell and macrocell users. For example one data point that has the value 3.2 indicates that on average the users received the quality representation 3.2. Hence, higher values indicate that the users received on average higher video quality representations. The data points in these figures correspond to different values of η. Also, the data points correspond to the average (mean) of all the measurements for 100 randomly generated topologies. The sample variance for this set the measurements is between 0.1 and 0.2 which is fairly small compared to the value of the mean and its difference between all the tested systems.

Video Quality. For this set of results we present the average video quality of the picocell users versus the average video quality of the users associated to the macrocell (only from the MBS to its associated users) for different constant values of η to illustrate the impact of different TDRP. The results for all systems can be seen in Fig. 4(a,b) for f=0.5. JTRAVQS is superior when compared to PFRA for high user density and low PBS density in Fig. 4(a). As the number of the

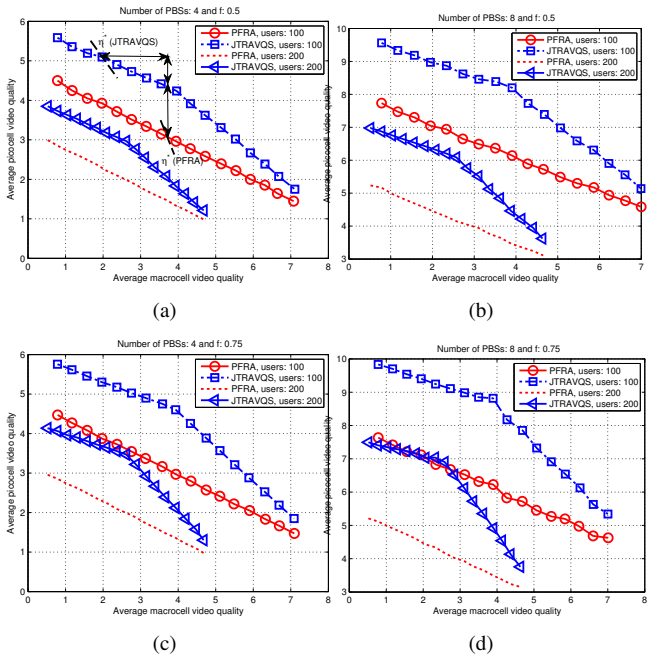

Fig. 4. Average macrocell vs. aggregate picocell video quality.

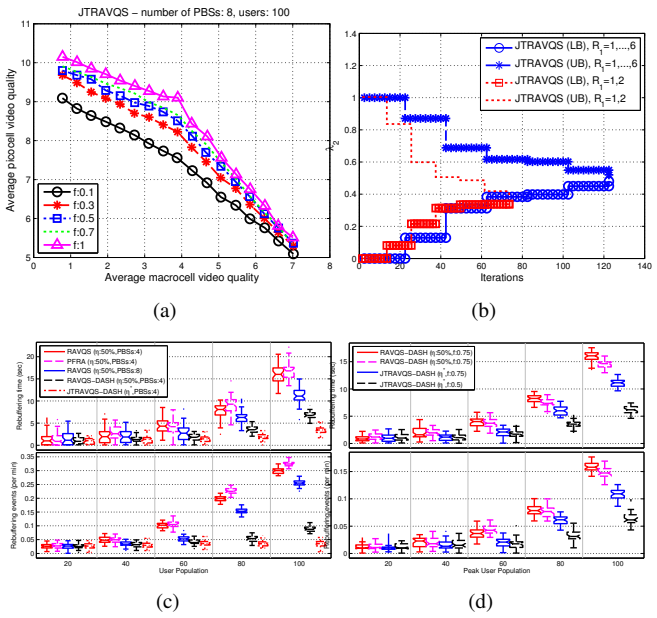

Fig. 5. (a) Video quality for different fraction of participating users. (b) Micro-benchmark for the upper bound (UB) and lower bound (LB) versus the number of iterations for JTRAVQS and a specific user. Two different sets of available video quality representations are shown ($|\mathcal{R}_1|$=2 and $|\mathcal{R}_1|$=6). (c) Rebuffering time/events vs. total number of users. (d) Rebuffering time/events vs. peak number of users with user churn.

PBSs is increased to 8 in Fig. 4(b), all the systems can achieve higher performance. The reason is that fewer users are associated to each picocell and so a higher communication rate is available for each user under any scheme. So more picocells leads to better results due to the higher available rate per user as expected. Another important result is that for constant PBS density (either 4 or 8), we have higher gain as the user population grows. The reason is the higher importance of optimal rate allocation, since the rate of a single PBS is shared among several users. For example in the very left data point of Fig. 4(b) performance improvement of JTRAVQS over PFRA is 21% for 100 users and 36% for 200 users). Of course the average video quality is reduced for all systems since more users are present. Also note that in the left part of the x axis, where all the resources are practically allocated to the picocells ($\eta \approx 1$), we observe the maximum possible video quality in the network. In this regime, the performance gap between JTRAVQS and the other systems is increased as the number of picocells and users is increased.

Another important result in the same figure, is related to the optimal η^*. It is indicated with a dashed line that is intersected with representative performance curves. This shows that the interpretation of the optimal TDRP with JTRAVQS, that is denoted as η^*(JTRAVQS), results in higher value for η^* when compared to η^*(PFRA) by 22% (highlighted with the horizontal arrow). Also if we assume that the system executes first PFRA to calculate the optimal TDRP indicated as η^*(PFRA), and then perform RAVQS [14] with this fixed value, the result is an average quality equal to 4.3 (the gain is highlighted with the lower vertical arrow). However, our complete system calculates the optimal operating point indicated with η^*(JTRAVQS) in the figure which gives an average quality equal to 5.1, a performance difference of 18.6% (the gain is highlighted with the upper vertical arrow).

Our system offers significant performance increase for f=0.5 but the benefits are more important when f=0.75 in Fig. 4(c,d). Note that for f=0.75 the slope of the curve is reduced as the η is decreased. The benefit is because we have a higher number of users that can be optimized under JTRAVQS. Also in this case the benefits are even more important when the fraction of the resources $1-\eta$ that the macrocell uses is below 50% (left part of the x axis) since this gives more resources to the highly spectral efficient links in the picocells to be used and so a higher communication rate is possible.

Fraction of Optimized Users. Now an interesting set of results is obtained for different values of f. We notice in Fig. 5(a) that as this fraction is reduced, JTRAVQS essentially degenerates to the PFRA system. Nevertheless, we still obtain significant benefits even for ratios of f around 30% since few users are enough for JTRAVQS to be able to improve the overall system performance.

Primal-Dual Convergence. The convergence speed of JTRAVQS versus the number of iterations is illustrated in Fig. 5(b). Results for the algorithm execution are shown for a specific fixed number of picocells and user population. The results for the primal-dual algorithm used for JTRAVQS show that the convergence is achieved within 150 iterations. Also Fig. 5(b) illustrates another important aspect of our system: If the system uses fewer discrete quality representations for each video file, it allows the faster convergence of the algorithm.

JTRAVQS-DASH Evaluation. The performance of JTRAVQS-DASH is evaluated with the same setup as before, that is however augmented when necessary. Now we present results for the playback performance: The time that a client is rebuffering in seconds, and the number of rebuffering events

per minute. To ensure fairness, we calculate first the optimal solution with JTRAVQS. Then, the minimum video quality for the JTRAVQS-DASH system is set equal to JTRAVQS. This ensures that JTRAVQS-DASH delivers at least the same video quality and the question is then to evaluate its ability to minimize rebuffering.[5]

For static user conditions the results are illustrated in Fig. 5(c). We draw a notched box plot of the rebuffering time for all the clients in the upper part of Fig. 5(c), and the number of rebuffering events in the lower part of Fig. 5(c). The notch here marks the 95% confidence interval for the *median*.[6] All the systems perform well for 20 and 40 users since high capacity is available in the network. However, for higher user densities the buffering time with the baseline JTRAVQS is increased by a factor that is worse than linear. The same is true for PFRA that is slightly worse. With a higher number of 8 PBSs, rebuffering time is improved for JTRAVQS because of higher available capacity. The same is true for the number of rebuffering events/minute that is a very high number for the first three systems we discussed (2-3 events in a 10 minute period). Hence, the higher capacity in the network achieved with 8 PBSs, simply delays the inevitable sharp increase but only of the rebuffering time. This result has an interesting interpretation for MNOs: With increasing user density, expanding the network with more small cells improves marginally the rebuffering time and practically not at all the rebuffering events. This is in contrast to the video quality that can achieve significant improvement in our earlier plots. For extra gains, solutions with buffer-awareness are required.

Better results are obtained for RAVQS-DASH again with a constant η=50%. This is effectively a system configuration that encompasses the main features of the DASH-aware streaming literature for single cell networks, e.g. [12], where the rate allocation and video quality are optimized by considering the buffer contents, but the overall communication resources are constant. Recall than η=50% is the optimal η under a PFRA metric for a population of 100 users and 4 PBSs (as illustrated in our earlier figures). Buffer-awareness can indeed reduce the rebuffering when compared to the previous systems, while it can also reduce the variations of playback buffering for the users (we have more predictable performance). The proposed JTRAVQS-DASH system illustrates that it can reduce the time spent in rebuffering by over 50% when compared to the previous system, while the number of rebuffering events is reduced even more significantly (1 event/25 min. vs. 1 event/10 min.). Hence, for a HCN the fixed TDRP is not the best option even if we design a DASH-aware system. Also, our overall results indicate that using a fixed TDRP has worse consequences in the rebuffering time/events of DASH than on the video quality (e.g., the results illustrated in Fig. 4).

Finally, we evaluate our system for a time-varying user population based on results from a real 3G network reported in [15]. We simulate an 8 hour period between 4pm and 12am, with a user peak occurring around 9pm [15]. During this peak, the number of users is nearly 30% higher than the number of users at 4pm and 12am. Hence, in our simulation we set accordingly $|\mathcal{N}_{jt}|$ (the increase and decrease are approximated as linear in time as shown in [15]). In the results in Fig. 5(d) we present in the x axis the peak number of users. C_{it} for each user i is also affected since TCP shares equally the communication rate among the competing traffic. Also, we only compare different flavors of JTRAVQS-DASH since the previous systems are not designed for a dynamic network. First, we observe again that the systems with fixed resource partitioning η=50% have worse performance. Second, we notice that the rebuffering time is higher when the fraction of optimized users is also high and equal to f=0.75. This means that with increasing user density in a network with user churn, optimizing a lower fraction f of the users increases the DASH playback quality of the optimized users by a significant amount for JTRAVQS-DASH. This is an important result since it provides a tool for an MNO to differentiate QoE in terms of rebuffering to various users.

[5]One can plot a synthetic metric of the two but this is not easily interpreted in terms of real QoE.

[6]Note that we plot the median here instead of the mean to avoid being influenced by outliers.

VI. Conclusions

In this paper, we presented a framework for improving the quality of video streaming in a HCN that employs TDRP. TDRP is essential for the efficient operation of HCNs and when high quality video distribution enters the game, efficiency becomes even more important. Our framework addressed precisely this challenge, i.e., it ensures optimal and video-aware allocation of resources in HCNs that apply TDRP. We formulated this problem in a linear non-convex formulation for which we proposed a primal-dual approximation algorithm. Our problem was decomposed into several problems that included a convex rate allocation problem, and a binary ILP for optimal video quality selection. An extensive performance evaluation under different HCN configurations highlighted the value of our framework for obtaining video quality improvements. Another implication of our solution approach is that it can be solved very fast. This allowed us to augment it to support the more challenging case of DASH. In this case significant additional improvement in terms playback performance was obtained.

References

[1] Ericsson Mobility Report: On the Pulse of the Networked Society, November 2014.

[2] N. Golrezaei, K. Shanmugam, A. Dimakis, A. Molisch, and G. Caire, "Femtocaching: Wireless video content delivery through distributed caching helpers," in *IEEE Infocom*, 2012.

[3] K. Poularakis, G. Iosifidis, A. Argyriou, and L. Tassiulas, "Video delivery over heterogeneous cellular networks: Optimizing cost and performance," in *IEEE Infocom*, 2014.

[4] 3GPP, "LTE-Advanced," http://www.3gpp.org/specifications/releases/68-release-12, 2013.

[5] S. Deb, P. Monogioudis, J. Miernik, and J. Seymour, "Algorithms for enhanced inter-cell interference coordination (eicic) in lte hetnets," *Networking, IEEE/ACM Transactions on*, vol. 22, no. 1, pp. 137–150, Feb 2014.

[6] S. Singh and J. Andrews, "Joint resource partitioning and offloading in heterogeneous cellular networks," *Wireless Communications, IEEE Transactions on*, vol. 13, no. 2, pp. 888–901, February 2014.

[7] D. Kosmanos, A. Argyriou, Y. Liu, L. Tassiulas, and S. Ci, "A cooperative protocol for video streaming in dense small cell wireless relay networks," *Signal Processing: Image Communication*, vol. 31, pp. 151–160, February 2015.

[8] S. Shakkottai and R. Srikant, "Network optimization and control," *Found. Trends Netw.*, vol. 2, no. 3, pp. 271–379, Jan. 2007.
[9] J. Chen, R. Mahindra, M. A. Khojastepour, S. Rangarajan, and M. Chiang, "A scheduling framework for adaptive video delivery over cellular networks," in *MobiCom*, 2013.
[10] A. Ahmedin, K. Pandit, D. Ghosal, and A. Ghosh, "Content and buffer aware scheduling for video delivery over lte," in *CoNEXT*, 2013.
[11] S. Singh, O. Oyman, A. Papathanassiou, D. Chatterjee, and J. Andrews, "Video capacity and qoe enhancements over lte," in *IEEE ICC*, 2012.
[12] V. Joseph and G. de Veciana, "Nova: Qoe-driven optimization of dash-based video delivery in networks," in *IEEE Infocom*, 2014.
[13] F. Dobrian, V. Sekar, A. Awan, I. Stoica, D. Joseph, A. Ganjam, J. Zhan, and H. Zhang, "Understanding the impact of video quality on user engagement," in *Proceedings of the ACM SIGCOMM Conference*, 2011.
[14] A. Argyriou, D. Kosmanos, L. Tassiulas, , Y. Liu, and S. Ci, "Video-aware time-domain resource partitioning in heterogeneous cellular networks," in *IEEE ICC*, 2015.
[15] S. Woo *et al.*, "Comparison of caching strategies in modern cellular backhaul networks," in *MobiSys*, 2013.
[16] W. Wu, A. Arefin, R. Rivas, K. Nahrstedt, R. Sheppard, and Z. Yang, "Quality of experience in distributed interactive multimedia environments: Toward a theoretical framework," in *ACM Multimedia*, 2009.
[17] J. Chakareski, J. Apostolopoulos, S. Wee, W. tian Tan, and B. Girod, "Rate-distortion hint tracks for adaptive video streaming," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 15, no. 10, pp. 1257–1269, Oct 2005.
[18] P. A. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," *MSR Technical Report MSR-TR-2001-35*, 2001.
[19] N. Freris, C.-H. Hsu, J. Singh, and X. Zhu, "Distortion-aware scalable video streaming to multinetwork clients," *Networking, IEEE/ACM Transactions on*, vol. 21, no. 2, pp. 469–481, April 2013.
[20] Video Trace Library: http://trace.eas.asu.edu/.
[21] Media presentation description and segment formats, http://mpeg.chiariglione.org/, Jan 2015.
[22] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.
[23] J. G. Proakis, *Digital Communications*, 4th ed. McGraw-Hill, 2000.
[24] A. Argyriou, "Error-resilient video encoding and transmission in multirate wireless lans," *Multimedia, IEEE Transactions on*, vol. 10, no. 5, pp. 691–700, Aug 2008.
[25] D. Bertsekas, A. Nedic, and A. Ozdaglar, *Convex Analysis and Optimization*. Athena Scientific Press, 2003.
[26] D. Palomar and M. Chiang, "A tutorial on decomposition methods for network utility maximization," *Selected Areas in Communications, IEEE Journal on*, vol. 24, no. 8, pp. 1439–1451, Aug 2006.
[27] "Small cell forum, backhaul technologies for small cells: Use cases, requirements and solutions," 2013.
[28] T. Stockhammer, "Dynamic adaptive streaming over http: Standards and design principles," in *ACM Multimedia Systems*, 2011.
[29] D. C. H. Nam, B. H. Kim, and H. G. Schulzrinne, "Mobile video is inefficient: A traffic analysis," Columbia University, Technical report, 2013.
[30] A. Mansy, M. Ammar, J. Chandrashekar, and A. Sheth, "Characterizing client behavior of commercial mobile video streaming services," in *Workshop on Mobile Video Delivery (MoVid)*, 2013.
[31] A. Argyriou, "Distortion-optimized scheduling of packetized video for Internet streaming with TCP," in *Packet Video Workshop*, Lausanne, Switzerland, November 2007.