Energy-Efficient Adaptive Video Transmission: Exploiting Rate Predictions in Wireless Networks

Hatem Abou-zeid, *Student Member, IEEE*, Hossam S. Hassanein, *Senior Member, IEEE*, and Stefan Valentin, *Member, IEEE*

***Abstract*—The unprecedented growth of mobile video traffic is adding significant pressure to the energy drain at both the network and the end user. Energy-efficient video transmission techniques are thus imperative to cope with the challenge of satisfying user demand at sustainable costs. In this paper, we investigate how predicted user rates can be exploited for energy-efficient video streaming with the popular Hypertext Transfer Protocol (HTTP)-based adaptive streaming (AS) protocols [e.g., dynamic adaptive streaming over HTTP (DASH)]. To this end, we develop an energy-efficient predictive green streaming (PGS) optimization framework that leverages predictions of wireless data rates to achieve the following objectives: 1) Minimize the required transmission airtime without causing streaming interruptions; 2) minimize total downlink base station (BS) power consumption for cases where BSs can be switched off in deep sleep; and 3) enable a tradeoff between AS quality and energy consumption. Our framework is first formulated as mixed-integer linear programming (MILP) where decisions on multiuser rate allocation, video segment quality, and BS transmit power are *jointly* optimized. Then, to provide an online solution, we present a polynomial-time heuristic algorithm that decouples the PGS problem into multiple stages. We provide a performance analysis of the proposed methods by simulations, and numerical results demonstrate that the PGS framework yields significant energy savings.**

***Index Terms*—Channel state prediction, dynamic adaptive streaming over HTTP (DASH), energy efficiency, mobility, resource allocation, wireless access networks.**

I. Introduction

INCREASING mobile data traffic and dense deployment of base stations (BSs) have made energy efficiency in networks imperative. This traffic growth is not only adding more pressure to the network and user device energy drain but also increases network operational expenditures (OPEXs) and negatively impacts the environment [1]. Consequently, research and standardization efforts are focusing on devising green mechanisms to save energy across the network. Among the wireless network elements, BSs account for more than 50% of the network energy consumption [2]. Reducing BS downlink transmit power by efficient rate allocation will thus result in monetary savings for the operator and reduce CO_2 emissions. Furthermore, energy-efficient rate allocation can improve the spectral efficiency and provide additional resources at high demand. Therefore, devising green radio *access* strategies is vital for overall network performance.

Meanwhile, video streaming is experiencing unprecedented growth with forecasts indicating that it will soon account for 66% of the total mobile traffic [3]. This is driven by the increasing capabilities of mobile devices and by content aggregation sites such as Netflix and YouTube. In particular, adaptive streaming (AS) is gaining popularity due to its ability to seamlessly adapt streaming quality to the current wireless data rate. In Hypertext Transfer Protocol (HTTP)-based implementations, such as HTTP live streaming [4], or dynamic AS over HTTP (DASH) [5], the video content is divided into a sequence of small file segments, each containing a short interval of playback time. Each segment is made available at multiple bit rates, and depending on the network conditions, the suitable segment quality is selected for transmission [6]. This reduces video freezing and is particularly suited for mobile video streaming where users experience channel gain fluctuations. While AS improves user quality-of-service (QoS), energy consumption still remains a fundamental challenge.

In this paper, we investigate how predictions of user rates can be exploited for energy-efficient transmission of stored videos that can be strategically buffered at the user devices. The predictability of a wireless channel is generally possible due to the correlation between location and channel capacity [7], [8]. Therefore, if a user's future location is known, the upcoming data rates can be anticipated from radio and coverage maps stored at the network, which can be also updated in real-time from user equipment (UE) measurements [9]. While mobility predictions are particularly plausible for users commuting in public transportation, trains, or vehicles on highways [10], studies on human mobility patterns reveal a high degree of temporal and spatial regularity [11], suggesting a potential 93% predictability [12]. A key motivation for incorporating such predictions is the plethora of navigation and context information available in today's smartphones.

Being aware of a user's upcoming rate allows the network to plan spectrally efficient rate allocations in advance without violating user streaming demands. For instance, if a user is moving toward the cell edge or a tunnel, the network can increase the allocated wireless resources, allowing the user to buffer more

Manuscript received August 21, 2013; revised January 12, 2014; accepted February 27, 2014. Date of publication April 1, 2014; date of current version June 12, 2014. The Associate Editor for this paper were the Guest Editors for the Special Section on Green Mobile Multimedia Communications.

H. Abou-zeid is with the Department of Electrical and Computer Engineering, Queen's University, Kingston, ON K7L 3N6, Canada (e-mail: h.abouzeid@queensu.ca).

H. S. Hassanein is with the School of Computing, Queen's University, Kingston, ON K7L 2N8, Canada (e-mail: hossam@cs.queensu.ca).

S. Valentin is with Bell Labs, Alcatel-Lucent, 70435 Stuttgart, Germany (e-mail: stefan.valentin@alcatel-lucent.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2014.2314646

video segments. *Prebuffering* these additional segments then provides smooth video streaming since the user can consume the buffer while being in poor radio coverage. Additionally, by not serving users in such conditions, the network-wide spectral efficiency increases since valuable channel resources can be provided to other users instead. If, on the other hand, the user is approaching the BS, transmission can be delayed, provided sufficient video segments have previously been buffered. This allows the BS to save energy by "sleeping" as the user approaches and then "waking up" for a short period, during which a high data transmission is possible.

We summarize the main contributions of this paper in the following.

- We propose a predictive green streaming (PGS) optimization framework that exploits rate predictions over a time horizon to 1) minimize the required transmission airtime subject to a target average quality level, without causing streaming interruptions; 2) minimize total downlink BS power consumption for cases where BSs can be switched off in deep sleep; and 3) enable a tradeoff between video quality and energy consumption.
- The PGS problem is formulated as an MILP that jointly determines multiuser resource allocation (RA), video segment quality levels, and BS on/off status. The proposed formulation captures 1) the joint relationship between *cumulative* user RA and *long-term* segment quality planning and 2) the load-dependent BS power consumption, with a minimum off duration for deep sleep modes.
- For online implementation, we present a polynomial-time algorithm that solves the PGS problem. Results demonstrate that the proposed algorithm stays close to the MILP benchmark in energy consumption, while exhibiting higher QoS *robustness* to rate prediction errors compared with the MILP.

We compare the performance and robustness of the PGS approaches through extensive simulation under realistic assumptions on cellular networks and vehicular mobility. We observe up to 85% energy reduction, while achieving comparable QoS, with respect to baseline solutions. Our results demonstrate that PGS is a promising energy-saving framework for future cellular networks.

The remainder of this paper is organized as follows. We review related work in Section II and introduce the system model in Section III. The MILP formulation of the PGS framework is developed in Section IV, whereas Section V presents the proposed PGS algorithm. In Section VI, we present simulation results to study the power consumption and video quality performance of PGS, and its robustness to prediction errors. Finally, conclusions are given in Section VII.

II. Background and Related Work

This paper addresses the problem of energy-efficient downlink transmission for adaptive video streaming, in a multicell network. Here, we first provide a background on *traffic-aware* energy-efficient radio access and then discuss related works that exploit rate predictions in detail.

A. Traffic-Aware Energy-Efficient Radio Access

As networks are over dimensioned to serve peak user demands, radio access energy can be reduced in a number of ways at times of lower demand. Such mechanisms include 1) *time-domain duty cycling* [2], [13] that utilizes only a fraction of the transmission slots and puts the BS at low energy operating modes during times of inactivity and 2) *frequency-domain duty cycling*, where only a fraction of the bandwidth (or physical resource blocks) is used for transmission [14]. Additionally, when demand is low for prolonged periods, BSs can be powered down to deep sleep modes that consume negligible power [15], [16]. Information on the temporal and spatial user traffic demand can assist networks to make better adaptations that reduce energy consumption. Such traffic awareness is incorporated in [17] where an optimal on/off-switching framework is developed to maximize energy saving under service constraints. More recently, in [18] and [19], multicell cooperation is proposed to configure the network layout by powering down select BSs depending on network traffic. While the preceding works focus on traffic-aware energy efficiency in general, they do not investigate using predictions to reduce BS utilization and energy.

B. Exploiting Mobility-Based Rate Predictions

The potential of *mobility-based* rate predictions is receiving increasing interest in recent literature. Ali *et al.* [20] showed how the system throughput can be increased with such predictions, whereas Abou-zeid and colleagues and Margolies *et al.* [21]–[23] discussed improvements in fairness as well under more realistic evaluation scenarios. Margolies *et al.* [23] also used extensive channel measurements from a third-generation (3G) network to show that a user's channel state is highly reproducible.

Leveraging rate predictions for wireless video streaming has been discussed in [24]–[26]. Yao *et al.* [24] developed a rate adaptation algorithm that proactively switches to the predicted transmission rates. This improves TCP rate control and throughput by faster convergence to the available capacity; however, it does not prebuffer segments or adapt quality based on predictions. This is addressed in [25] and [26], where users heading to poor conditions request additional segments in advance. A prototype is presented in [26] that logs receiver bandwidth-location information to perform long-term bit-rate planning and prevent streaming disruptions. While these works assume that the user trajectory is known, a related *geo-predictive* quality adaptation mechanism for DASH has also recently been presented in [27], where the user path is also predicted. In [28], minimizing video interruptions by exploiting rate predictions has been studied. However, the focus of this work is on optimizing multiuser RA and not on adapting video quality, as in [24]–[27], where each client controls its bit-rate plan independently. This *in-network* RA facilitates obtaining network-wide objectives and efficiently trading off video quality among multiple users. Several recent resource management approaches for video streaming have also been proposed in [29]–[31], but predictions are not considered therein. The aforementioned works focus on enhancing user experience but do not address energy efficiency.

TABLE I
SUMMARY OF IMPORTANT SYMBOLS

Symbol	Description
$b_{k,n}$	Binary decision variable for on/off status of BS k at slot n
$\mathrm{BS}^{\mathrm{air}}_{k,n}$	Available airtime of BS k at slot n
$D_{i,s}$	Cumulative number of bits required by user i to stream the first s segments [bits]
i	User index, $i = \{1, 2, \cdots, M\}$
k	BS index, $k = \{1, 2, \cdots, K\}$
$\mathcal{K}$	Set of BSs in the network
$\mathcal{M}$	Set of users in the network
n	Time slot index, $n = \{1, 2, \cdots, N\}$
N	Number of slots in the lookahead window
N_{seg}	Number of slots in a video segment
$\mathcal{N}$	Set of time slots in the lookahead window
$\mathcal{N}^s$	Set of time slots belonging to segment s
$p_{k,n}$	Transmit power of BS k at slot n
$q_{i,s,l}$	Binary variable for quality level l of segment s for user i
$q_{\max}$	Maximum quality level
$\hat{r}_{i,n}$	Link rate of user i at slot n [bits]
$R_{i,n}$	Cumulative rate allocated to user i by slot n [bits]
s	Segment index, $s = \{1, 2, \cdots, S\}$
S	Number of segments in the lookahead window
T	Duration of the lookahead window [s]
τ	Duration of a time slot [s]
τ_{seg}	Duration of a video segment [s]
$\mathcal{U}_{k,n}$	Set containing the indices of users associated with BS k at slot n
$x_{i,n}$	Fraction of airtime assigned to user i at slot n

In [32]–[34], the work that was done is closest to this paper, where the primary objective is to exploit predictions for *energy efficiency*. Rate predictions are used to minimize system utilization and avoid streaming delays of constant bit-rate videos in [32]. Lu and de Veciana present a detailed buffer model and formulate the multiuser single-cell case as a nonconvex problem. Then, optimal algorithms for the single-user case are developed, and significant reductions in BS resource utilization are observed. In [33], we also discuss the potential energy saving that can be achieved by a *mobility-aware* wireless access framework. An architecture is presented with the composite functional elements, and their interaction is discussed. In [34], the problem of trading off video degradation with BS power consumption is considered, and predictive algorithms to solve the problem are presented. This paper differs from these works in several aspects. First, we now consider the delivery of adaptive video streams, and thus model and solve the *joint* rate allocation and quality planning problem over a time horizon. Second, in addition to saving power by minimizing utilization, we also incorporate deep sleep modes where BSs can be switched off. Finally, we formulate a detailed multiuser multicell optimization framework for energy-efficient AS and present an efficient heuristic algorithm to solve the problem.

III. SYSTEM MODEL AND PRELIMINARIES

Here, we present the system model and assumptions. We use the following notational conventions: $\mathcal{X}$ denotes a set, and its cardinality $|\mathcal{X}|$ is denoted X. We use bold letters to denote matrices, e.g., $\mathbf{x} = (x_{a,b} : a \in \mathbb{Z}_+, b \in \mathbb{Z}_+)$. $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ are the floor and ceiling functions, respectively. Frequently used notation is summarized in Table I.

A. Network Overview

Consider a network with BS set $\mathcal{K}$ and active user set $\mathcal{M}$. Users request stored video content that is transmitted using adaptive bit-rate streaming over HTTP. Time is divided in slots of equal duration τ, during which the wireless channel can be shared arbitrarily among multiple users. We assume that the wireless link is the bottleneck; therefore, the requested video content is always available at the BS for transmission.

B. Link Model and Resource Sharing

We assume a block fading model where the achievable data rate is assumed constant during each time slot. As we are interested in slow-fading variations, a typical value of such a coherence time τ is 1 s for vehicle speeds up to 20 m/s, during which average wireless capacity is not significantly affected. The achievable data rate depends on the path-loss model $\mathrm{PL}(d) = 128.1 + 37.6 \log_{10} d$, where the user–BS distance d is in kilometers [35]. The feasible link rate is computed using Shannon's equation with SNR clipping at 20 dB to account for practical modulation orders. Therefore, user i at slot n will have a feasible data rate of

$$r_{i,n} = \tau B \log_2 \left(1 + \frac{P_{\mathrm{rx}_{i,n}}}{N_o B}\right) \quad [\mathrm{bits}] \tag{1}$$

where P_{rx}, N_o, and B are the received power, noise power spectral density, and the transmission bandwidth, respectively.[1] Note that the slot user rate $r_{i,n}$ gives the number of bits that can be transmitted during a time slot, i.e., the transmission rate normalized with slot duration τ.

User link capacities are assumed known for the upcoming T seconds, which we call the *look-ahead window*. There are $N = T/\tau$ time slots within the look-ahead window, as shown in Fig. 1(a), which we denote the set $\mathcal{N} = \{1, 2, \ldots, N\}$. The future link capacities are determined by computing $P_{\mathrm{rx}_{i,n}}$

[1]This path-loss-dependent link model is an abstraction of a radio environment map that will, in practice, be available at the service provider.

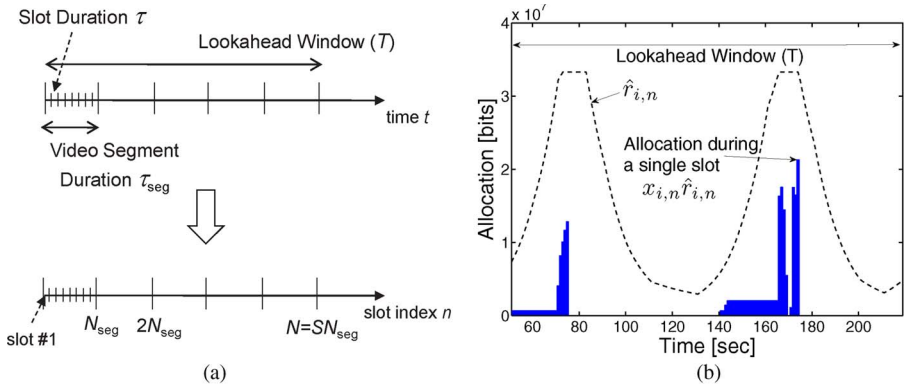

Fig. 1. System models and notation. (a) Defined time durations and slot indexes. (b) Sample user rate allocation during T.

based on the knowledge of the future user locations and then by substituting in (1). This will generate a matrix of future link rates as defined by $\hat{\mathbf{r}} = (\hat{r}_{i,n} : i \in \mathcal{M},\ n \in \mathcal{N})$. Fig. 1(b) shows an example of $\hat{r}_{i,n},\ \forall n$, for a user traversing two BSs along a highway. We first assume that knowledge of $\hat{\mathbf{r}}$ is error free to provide the bounds of the potential gains and then assess the impact of prediction errors on the gains.

Fig. 1(a) also shows the video segment duration τ_{seg}, which is a multiple of τ. The look-ahead window T is also selected to be divisible by τ_{seg}, as shown in the figure. In terms of time slots, $N_{\text{seg}} = \tau_{\text{seg}}/\tau$ denotes the number of slots that make up one video segment, and $S = N/N_{\text{seg}}$ denotes the number of video segments during T.

BS airtime is shared among the active users during each slot n. We define the rate-allocation matrix $\mathbf{x} = (x_{i,n} \in [0,\ 1] : i \in \mathcal{M}, n \in \mathcal{N})$, which gives the fraction of time during each slot n that the BS bandwidth is assigned to user i. The rate received by each user at each slot is the element-wise product $\mathbf{x} \odot \hat{\mathbf{r}}$. Therefore, $\mathbf{x}$ controls both the *per-slot* and total *long-term* rates users receive over the N slots. A sample allocation $x_{i,n},\ \forall n$, for a user i is shown in Fig. 1(b), where the bars indicate the proportion of $\hat{r}_{i,n}$ allocated to that user. Note that, since a user can traverse multiple cells during N, BS cooperation is needed to make the allocation plan. This is assumed possible via an inter-BS interface such as the X2-interface in 3G Partnership Project compliant networks. User–BS association is based on the strongest received signal. We can define the set $\mathcal{U}_{k,n}$, $k \in \mathcal{K},\ n \in \mathcal{N}$, which contains the indexes of all the users associated with BS k at slot n.

C. Adaptive Video Streaming Model

In AS over HTTP, the video content is divided into a sequence of small HTTP-based file segments. Video segments are then preencoded in multiple versions, each with a specific video bit rate and resolution or "quality level" [36]. Higher quality segments will be larger in size but represent similar playback duration. We denote the segment quality levels by $l \in \mathcal{Q}$, where $\mathcal{Q} = \{1, 2, \ldots, q_{\max}\}$, and $q_{\max}$ is the maximum quality level. The function $f^{Q}_{\text{rate}}(\cdot)$ maps the quality level to the corresponding bit rate. Higher segment qualities will require higher bit rates for successful reception; therefore, $f^{Q}_{\text{rate}}(\cdot)$ is an increasing function of l. To assign the quality level of each user segment, we define the binary decision array $\mathbf{q} = (q_{i,s,l} \in \{0,\ 1\} : i \in \mathcal{M},\ s = \{1, 2, \ldots, S\},\ l \in \mathcal{Q})$. If there are three quality levels and $q_{i,s,1} = 1$, then user i will receive segment s at quality level 1, and the remaining quality level indexes are zero, i.e., $q_{i,s,2} = 0$ and $q_{i,s,3} = 0$. Therefore, to ensure that only one level is selected, $\sum_{l=1}^{q_{\max}} q_{i,s,l} = 1,\ \forall i, s$.

D. BS Power Consumption Model

The BS downlink power consumption is based on the linear load-dependent power model [14], [37], where power is proportional to the BS load, with a fixed power required at minimum load. For BS k at slot n, this can be represented as

$$p_{k,n} = \begin{cases} P_0 + (P_m - P_0)\text{BS}^{\text{load}}_{k,n}, & 0 < \text{BS}^{\text{load}}_{k,n} \leq 1 \\ P_{\text{sleep}}, & \text{BS}^{\text{load}}_{k,n} = 0 \end{cases} \tag{2}$$

where P_m and P_0 are the power consumption at the maximum and minimum nonzero loads, and the BS load is computed as $\text{BS}^{\text{load}}_{k,n} = \sum_{i \in \mathcal{U}_{k,n}} x_{i,n}$. When there is no load, the BS can enter a sleep mode, which consumes $P_{\text{sleep}}[W]$. Advanced BS hardware allows P_{sleep} to be a fraction of P_0, or even zero, thus allowing a complete BS switch off [14]; therefore, we assume $P_{\text{sleep}} = 0$. BSs entering this deep sleep mode are required to remain off for at least n_{off} time slots to allow sufficient time before a wake up is possible. We denote the BS power per slot matrix by $\mathbf{p} = (p_{k,n} \in [0,\ P_m] : k \in \mathcal{K},\ n \in \mathcal{N})$ and the BS on/off binary decision matrix by $\mathbf{b} = (b_{k,n} \in \{0, 1\} : k \in \mathcal{K}, n \in \mathcal{N})$.

IV. Predictive Green Streaming Framework

Here, we present the PGS framework that leverages rate prediction to minimize BS power consumption and transmission time of adaptive video streams. As opposed to live streaming, stored videos can be strategically delivered ahead of time and cached at the UE, after which transmission can be momentarily suspended while the user consumes the buffer. The essence of PGS is that a *long-term* RA plan is made for each user by exploiting its individual rate predictions. By allocating over a

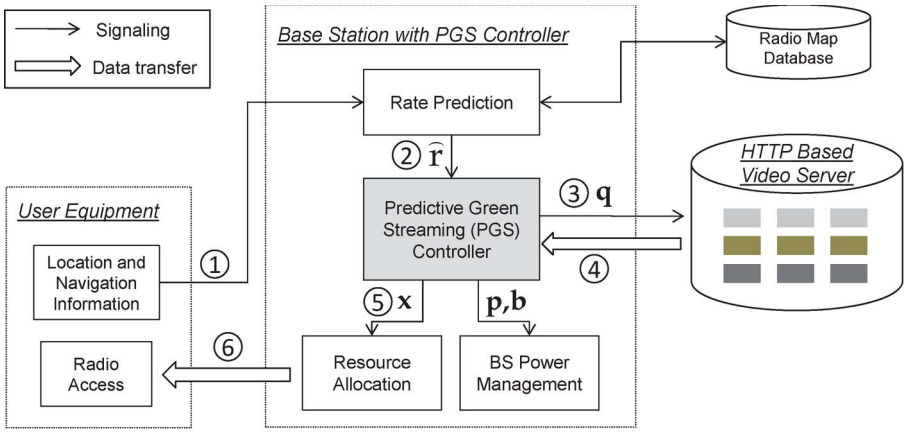

Fig. 2. PGS operation.

time horizon, PGS can plan to grant more resources to users at their respective high data rates to prebuffer content efficiently and reduce transmission energy.

A. System Overview

In Fig. 2, we present the considered architecture for HTTP-based AS in the wireless network and outline the required steps in PGS to conceptualize its operation. First, we assume that user location and navigation information is provided to the BS, which determines the future rates $\hat{\mathbf{r}}$ that users will experience by consulting a radio map database. Since our focus is to develop the predictive AS transmission schemes, we assume $\hat{\mathbf{r}}$ is provided to the PGS controller, and thereafter, we investigate the effect of prediction errors in Section VI. The PGS framework uses $\hat{\mathbf{r}}$ defined over a time horizon, to minimize power consumption while achieving a target video quality level with no video stalls. To do so, it jointly determines 1) the optimal user rate allocations $\mathbf{x}$, 2) the optimal video segment qualities $\mathbf{q}$, 3) the optimal BS transmit power $\mathbf{p}$, and 4) the BS on/off statuses $\mathbf{b}$. The required segments, as specified in $\mathbf{q}$, are requested from the HTTP-based video servers. These segments are then delivered to users over time slots in accordance with the determined rate allocation plan in $\mathbf{x}$. The PGS controller also determines the deep sleep schedule of the BSs that minimizes power consumption without violating user requirements through the optimization variable $\mathbf{b}$, which is passed onto the BS power management unit.

It is worth noting that, currently, DASH relies on the client to signal the requested quality levels to the content server [36]. Therefore, the proposed in-network PGS approach requires some modifications to traditional DASH operation. However, there are current efforts toward enabling forms of network assistance in DASH, under the MPEG server-and-network-assisted DASH (SAND) operation [38].

We formulate two objectives of PGS as MILPs to provide benchmark solutions for performance evaluation. The first objective, i.e., PGS-MinPower, minimizes total BS power consumption, where BSs can enter deep sleep, under the constraint that no users experience video stalling. The MILP formulation is nontrivial due to the tight coupling between the large number of optimization variables. We then present PGS-MinAir with the objective of minimizing transmission airtime. However, in PGS-MinAir, BS turn off is not enabled and can be considered a special case of PGS-MinPower. To formulate these problems, several constraints have to be considered, which can be classified into 1) user requirement constraints and 2) BS operation constraints.

B. User Constraints

1) Rate Allocation for Smooth Streaming: Consider a user streaming a stored video at the *maximum* quality level $q_{\max}$. For the video to playback without interruptions, $f^Q_{\text{rate}}(q_{\max})$ bits are required per second. Alternatively, a bulk of video content can be transmitted at once and buffered at the user's device, after which transmission can be suspended momentarily without causing video stalls. Therefore, we are interested in the cumulative video content stored at the user's device, which is given by $\sum_{n'=1}^{n} x_{i,n'}\,\hat{r}_{i,n'}$ at time slot n. With the knowledge of $\hat{\mathbf{r}}$, an allocation plan can be made that grants minimum resources when the channel conditions are low, and prebuffers as much content as possible when conditions are high. This will reduce transmission time, thereby reducing BS load and saving power. To illustrate the idea, Fig. 3(a) and (b) shows the difference between traditional allocation and the aforementioned predictive scheme where allocation is avoided during poor channel conditions.

The joint relationship between the cumulative allocated rate and segment quality selection that ensures smooth playback in AS is captured in the constraint

$$\tau_{\text{seg}} \sum_{s'=1}^{s} \sum_{l=1}^{q_{\max}} q_{i,s',l}\, f^Q_{\text{rate}}(l) \leq \sum_{n'=1}^{sN_{\text{seg}}} x_{i,n'}\,\hat{r}_{i,n'} \quad \forall\, i; \quad \forall\, s \tag{3}$$

$$\sum_{l=1}^{q_{\max}} q_{i,s,l} = 1 \qquad \forall\, i \in \mathcal{M}; \quad \forall\, s \in \{1,2,\ldots,S\} \tag{4}$$

where (4) ensures that one quality level is selected. The right-hand side (RHS) of (3) denotes the cumulative bits allocated to user i at the slots corresponding to the end of each segment,

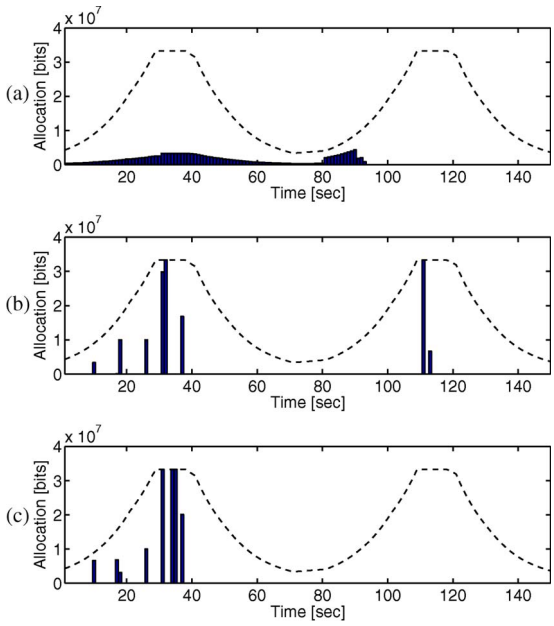

Fig. 3. Sample user allocation with time illustrating power minimization. (a) In traditional allocation, airtime is divided among users with per slot fairness considerations. (b) In PGS-MinAir, a user is preallocated video segments when the rate is high to avoid inefficient allocation during low-rate periods. (c) PGS-MinPower is similar to (b) with the additional objective of grouping user allocations to allow BSs to subsequently turn off.

whereas the left-hand side (LHS) expresses the cumulative bits *required* to download up to s video segments at the quality levels specified by $q_{i,\,s,\,l}$. For uninterrupted playback, an arbitrary segment s must be completely downloaded, by time slot sN_{seg}. Note that (3) allows a tradeoff between video quality and required airtime, while ensuring smooth playback.

2) Target Quality: If $l_{\text{req}} \in \{1, \ldots, q_{\max}\}$ denotes the desired average quality level for each user, then

$$\sum_{s=1}^{S} \sum_{l=1}^{q_{\max}} q_{i,\,s,\,l} \geq l_{\text{req}} S \qquad \forall\, i \in \mathcal{M} \tag{5}$$

represents the average user quality constraint.

3) User Buffer Limit: In addition to the key constraints in (3) and (5), a limit can be also imposed on the number of bits that can be prebuffered at the user's device. This may be due to the video client and network policy, or device limitations. If L_i denotes the limit for user i, then we have the constraint

$$\sum_{n'=1}^{n} x_{i,n'}\, \hat{r}_{i,n'} - \tau_{\text{seg}} \sum_{s'=1}^{\lfloor n/N_{\text{seg}} \rfloor} \sum_{l=1}^{q_{\max}} q_{i,s',l}\, f^{Q}_{\text{rate}}(l)$$

$$- \frac{\tau_{\text{seg}}}{N_{\text{seg}}}\, (n \bmod (N_{\text{seg}})) \sum_{l=1}^{q_{\max}} q_{i,\lceil n/N_{\text{seg}} \rceil,l}\, f^{Q}_{\text{rate}}(l) \leq L_i \quad \forall\, i;\, \forall\, n. \tag{6}$$

The LHS of (6) determines the difference between the cumulative allocated bits and the bits required for smooth playback at every slot n and therefore denotes the buffered bits. The second term on the LHS accounts for the bits of previously played video segments, and the third term represents the portion of the current segment that has been played.

C. BS Constraints

1) BS Resource Limit: The BS resource constraint limits the sum of the user airtime fractions to unity, i.e.,

$$\sum_{i \in \mathcal{U}_{k,\,n}} x_{i,\,n} \leq 1 \qquad \forall\, k \in \mathcal{K}; \quad \forall\, n \in \mathcal{N}. \tag{7}$$

This constraint is applied at each BS, where the summation is over all users i associated with BS k at slot n.

2) BS Slot Power Consumption: According to the BS power model of (2), the power consumed by each BS is dependent on 1) the total user airtime and 2) whether the BS is kept on or switched off. This is expressed by the following constraint:

$$(P_m - P_0) \sum_{i \in \mathcal{U}_{k,\,n}} x_{i,\,n} - p_{k,\,n} + P_0\, b_{k,\,n} = 0$$

$$\forall\, k \in \mathcal{K}; \quad \forall\, n \in \mathcal{N} \tag{8}$$

where the binary decision variable $b_{k,\,n}$ is multiplied by P_0 to produce zero-power consumption when the BS is off.

3) BS On Constraint: To enforce a BS to be on if there is any load, we apply the following constraint:

$$\sum_{i \in \mathcal{U}_{k,\,n}} x_{i,\,n} - b_{k,\,n} \leq 0, \qquad \forall\, k \in \mathcal{K}; \quad \forall\, n \in \mathcal{N}. \tag{9}$$

4) BS Off Indicator: To monitor when a BS is turned off, we introduce an indicator variable $I_{k,\,n}$ that is equal to 1 only when a BS is turned off. This is achieved by

$$-b_{k,\,n-1} + b_{k,\,n} + I_{k,\,n} = 0 \qquad \forall\, k \in \mathcal{K}; \quad \forall\, n \in \mathcal{N} \tag{10}$$

where $b_{k,\,0} = 0\ \forall\, k$. On the other hand, when a BS is switched on, $I_{k,\,n} = -1$, and if there is no change, $I_{k,\,n} = 0$. The value of this indicator is used by the following constraint to ensure that a BS remains off for a minimum number of n_{off} slots.

5) Minimum Off Time: To model the minimum off duration, we restrain the BS from turning on for n_{off} slots once it has been turned off. This can be achieved by

$$b_{k,\,n+c} + I_{k,\,n} \leq 1 \qquad \forall\, k \in \mathcal{K}; \quad \forall\, n \in \mathcal{N}; \quad \forall\, c \tag{11}$$

where $c = 1, \ldots, n_{\text{off}}$, and $n + c <= N$. Equation (11) ensures that if the indicator of the previous time slot is 1, then $b_{k,\,n+c}$ will have to remain equal to zero for n_{off} slots. This is controlled through the variable c that generates n_{off} constraints to define the on/off status of the upcoming n_{off} slots, for every n. If on the other hand, the indicator is not 1, then $b_{k,\,n+c}$ can take on any value.

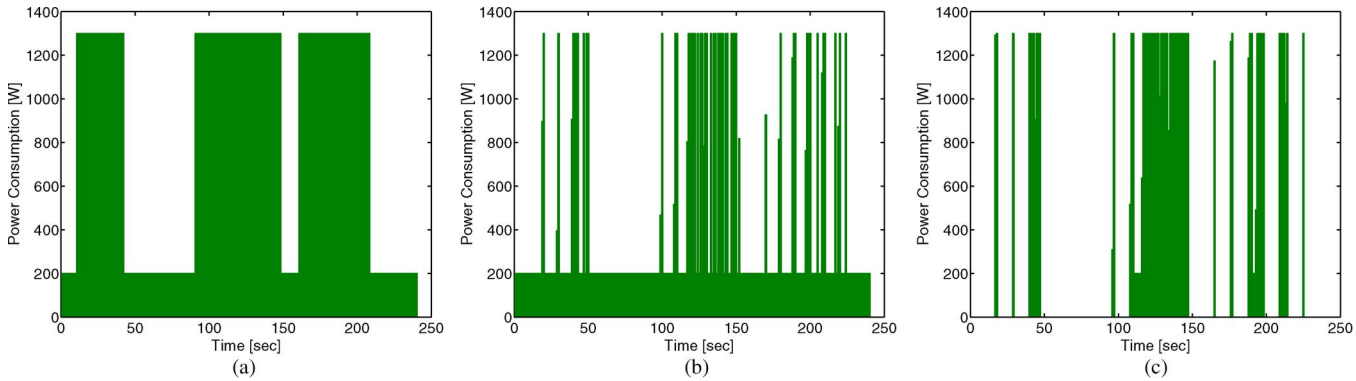

Fig. 4. Sample BS power consumption over time, where $P_0 = 200$ W and $P_m = 1300$ W. (a) In traditional operation, BS airtime is fully utilized when there are users present. (b) With PGS-MinAir, BS airtime is minimized by opportunistic allocations using rate predictions, as shown in Fig. 3(b). In (c), PGS-MinPower allows the BS to enter deep sleep modes and conserve more power. To do so, user allocations are grouped together, as shown in Fig. 3(c).

D. PGS-MinPower MILP Problem Definition

Considering the previously discussed constraints, the PGS-MinPower problem can be formulated as the following MILP:

$$\underset{\mathbf{x},\mathbf{q},\mathbf{p},\mathbf{b}}{\text{minimize}} \quad \sum_{k=1}^{K}\sum_{n=1}^{N} p_{k,n} \tag{12}$$

subject to : Constraints (3) to (11)

$$0 \le x_{i,n} \le 1 \qquad \forall i \in \mathcal{M}; \quad \forall n \in \mathcal{N}$$

$$q_{i,s,l} \in \{0,1\} \quad \forall i \in \mathcal{M}; \quad \forall s \in \mathcal{S}; \forall l \in \mathcal{Q}$$

$$0 \le p_{k,n} \le P_m \quad \forall k \in \mathcal{K}; \quad \forall n \in \mathcal{N}$$

$$b_{k,n} \in \{0,1\} \qquad \forall k \in \mathcal{K}; \quad \forall n \in \mathcal{N}.$$

Note that, although (12) provides the optimal joint allocations of all the decision variables, it is computationally intractable to solve large instances of PGS-MinPower due to the large number of $MN + MS + 2KN$ decision variables, and the coupling between them. Further, considerable memory is needed as the resulting constraint matrix has a size of $M + MN + 2MS + 5KN$, which can be very large. The duration of the look-ahead window impacts both N and S; therefore, the complexity of (12) depends primarily on the prediction window duration.

It is worth noting that overhead may be introduced when turning BSs off/on. This may be accounted for by increasing the value of n_{off} to prevent frequent short sleeps. Another way to directly incorporate this overhead is through the BS off indicator variable $I_{k,n}$ defined in (10). This can be achieved by adding another power consumption term to the objective in (12), which sums over $I_{k,n}$ multiplied by a constant that denotes the power consumption of a single on/off switch. The PGS solution will then minimize the total power consumed while accounting for the overhead of the deep sleep switches.

E. PGS-MinAir MILP Problem Definition

The PGS-MinAir problem considers the case where BSs cannot be switched off into deep sleep modes, for example, due to other types of traffic in the network. PGS-MinAir can be formulated by setting the BS on/off decision variable to 1 and excluding constraints (9)–(11) in (12). However, a more compact formulation can also exclude the BS power $p_{k,n}$ variables, and airtime can be minimized directly through user allocations $x_{i,n}$. This is possible since BS power is proportional to airtime in the linear BS power model of (2). Therefore, the PGS-Airtime problem can be formulated as

$$\underset{\mathbf{x},\mathbf{q}}{\text{minimize}} \quad \sum_{i=1}^{M}\sum_{n=1}^{N} x_{i,n} \tag{13}$$

subject to : Constraints (3) to (7)

$$0 \le x_{i,n} \le 1, \quad \forall i \in \mathcal{M}; \quad \forall n \in \mathcal{N}$$

$$q_{i,s,l} \in \{0,1\}, \quad \forall i \in \mathcal{M}; \quad \forall s \in \mathcal{S}; \forall l \in \mathcal{Q}.$$

Fig. 4 shows an example of the resulting BS power consumption plan for PGS versus a traditional scheme, where BS airtime is shared equally among video streaming users. In the scenario considered, vehicular users arrive at the BS in three consecutive groups. In Fig. 4(a), as long as users are present, BS airtime is completely utilized. However, in Fig. 4(b) and (c), PGS allows the BS to transmit in a spectrally efficient way without violating user streaming requirements. Note that, although PGS-MinAir minimizes total transmit time, PGS-MinPower in Fig. 4(c) is able to strike the optimal tradeoff between serving users when their individual rates are high and grouping user transmissions together (even if not at their respective best rates) to generate blocks of sleep time. This comes at the cost of increased complexity as observed in the PGS-MinPower formulation, where the optimization variables are tightly coupled. However, at high load, the power-saving gains of PGS-MinPower over PGS-MinAir will decrease and eventually converge to PGS-MinAir. This is due to the decreased ability to generate silent space long enough for a BS switch off. We discuss more details of the tradeoffs involved in the numerical results of Section VI. Before that, we present a polynomial-time solution of the PGS the problem, which achieves close to optimal results.

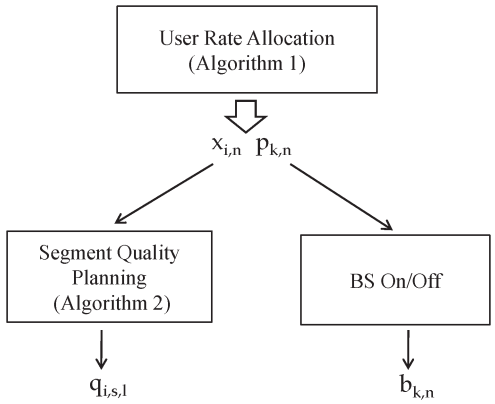

Fig. 5. Proposed multistage PGS solution.

V. Multistage Predictive Green Streaming Solution

Here, we develop a multistage approach to solve the PGS MILPs presented in Section IV. Fig. 5 outlines the steps involved. The core stage is a user rate-allocation algorithm that assigns BS airtime to users over the look-ahead window, thereby solving $\mathbf{x}$ and $\mathbf{p}$. Thereafter, segment qualities are explicitly planned for each user based on the allocated bits, and BS on/off statuses are determined from the resulting idle time in $\mathbf{p}$. This decoupling is based on the intuition that an efficient rate-allocation scheme exploiting rate predictions will provide power savings while satisfying user quality needs. Before discussing each stage, we introduce the following definitions.

- Cumulative demand $D_{i,s}$ is the total number of bits required by user i to stream the first s segments. For a given target quality level l_{req}, $D_{i,s} = s\, f^{Q}_{\text{rate}}(l_{\text{req}})$ [bits] $\forall\, i$.
- Cumulative rate allocation $R_{i,n}$ is the total rate allocated to user i by time slot n, i.e., $R_{i,n} = \sum_{n'=1}^{n} x_{i,n'} \hat{r}_{i,n'}$ $\forall\, i$.
- User rate percentile $\hat{r}^{y\%}_{i,n}$ is the yth percentile of the future user rate, i.e., computed over $\hat{r}_{i,n\le n'\le N}$ for each user.

A. Rate Allocation

The RA strategy is to divide the problem into a series of allocation subproblems performed at the start of each segment. The idea of this decomposition is to minimize airtime while focusing on satisfying the streaming constraint in (3), whichXis performed for each segment. If $\mathcal{N}^s$ denotes the set of slots comprising segment s, then $\mathcal{N}^s = \{(s-1)N_{\text{seg}}+1, (s-1)N_{\text{seg}}+2, \ldots, sN_{\text{seg}}\}$, and allocation is made incrementally for each $\mathcal{N}^s$. Each allocation is further divided into two steps: 1) airtime minimization and 2) opportunistic prebuffering. In the first step, users that do not have enough content prebuffered to stream the upcoming segment at the target quality level are prioritized, and their demands are fulfilled with the minimum airtime. In the second step, users that have exceptionally good channel conditions are granted excess airtime to prebuffer future video content. This will reduce the airtime required later to download upcoming segments. Next, we discuss the details of each step.

1) Airtime Minimization: At the start of segment s, each BS k determines the set of priority users $\mathcal{P}_{k,s}$ that have insufficient cumulative allocation to play the upcoming video segment at the target quality level. Set $\mathcal{P}_{k,s}$ will therefore not include users that have prebuffered segments. The required rate allocation $r^{\mathcal{P}}_i$ for user i can be then computed as

$$r^{\mathcal{P}}_i = D_{i,s} - R_{i,sN_{\text{seg}}} \qquad \forall\, i \in \mathcal{P}_{k,s} \tag{14}$$

where $R_{i,0} = 0,\ \forall\, i$. To serve the users with these rate requirements, using the minimum BS airtime, we need to solve the optimization problem

$$\begin{aligned}
\underset{\mathbf{x},\,\mathbf{Y}}{\text{minimize}} \quad & \sum_{n\in\mathcal{N}^s}\sum_{i\in\mathcal{P}_{k,s}} x_{i,n} + \beta \sum_{i\in\mathcal{P}_{k,s}} Y_i \qquad (15)\\
\text{subject to :} \quad & -\sum_{i\in\mathcal{P}_{k,s}} x_{i,n}\hat{r}_{i,n} - Y_i \le -r^{\mathcal{P}}_i \quad \forall\, n \in \mathcal{N}^s\\
& \sum_{i\in\mathcal{P}_{k,s}} x_{i,n} \le 1 \qquad \forall\, n \in \mathcal{N}^s\\
& 0 \le x_{i,n} \le 1 \qquad \forall\, i \in \mathcal{P}_{k,s},\ n \in \mathcal{N}^s\\
& 0 \le Y_i \qquad \forall\, i \in \mathcal{P}_{k,s}.
\end{aligned}$$

Variables Y_i are introduced to capture any unfulfilled rate allocation, when it is not possible to satisfy all user requirements. As satisfying users with the target quality level has higher precedence over saving airtime, weight parameter $\beta > 1$. Generally, at low-to-moderate loads (where there are potential power savings), (15) will yield $Y_i = 0$, and quality requirements will be met with minimum BS airtime.

Note that the problem in (15) has a linear objective function with linear constraints and is therefore a linear program (LP), which can generally be solved efficiently, even with the widely used Simplex algorithm [39]. Further, the problem dimension is incomparable to the optimal PGS MILP formulations and therefore provides significant computational and memory requirement gains.

Alternatively, to avoid the requirement of BSs being equipped with LP solvers, we present the following simple algorithm to solve (15). First, the set $\mathcal{P}_{k,s}$ is sorted in descending order of user requirements $r^{\mathcal{P}}_i$. Then, each user $i \in \mathcal{P}_{k,n}$ is selected in sequence to transmit at the time slot $n^* \in \mathcal{N}^s$ that has the largest predicted rate for that user, i.e.,

$$n^* = \arg\max_{n} \hat{r}_{i,n} \qquad \forall\, n \in \mathcal{N}^s. \tag{16}$$

Note that, if $\text{BS}^{\text{air}}_{k,n}$ denotes the airtime available in BS k at slot n, then the search in (16) will exclude slots where $\text{BS}^{\text{air}}_{k,n} = 0$. The airtime allocated to the user is then $x_{i,n^*} = \hat{r}_{i,n^*}/r^{\mathcal{P}}_i$, and the remaining BS airtime is updated to $\text{BS}^{\text{air}}_{k,n^*} = \text{BS}^{\text{air}}_{k,n^*} - x_{i,n^*}$. After iterating over all $i \in \mathcal{P}_{k,n}$, we update $R_{i,n}$, $\mathcal{P}_{k,s}$, and $r^{\mathcal{P}}_i$, and the process is repeated until either $\mathcal{P}_{k,s} = \phi$ or there is no remaining BS airtime for $n \in \mathcal{N}^s$. This procedure is outlined in lines 6–14 in Algorithm 1, and numerical results in

Section VI indicate that it provides almost identical results to the LP of (15).

Algorithm 1 User Rate Allocation Algorithm

Require: $\hat{r}_{i,n}, \mathcal{U}_{k,n}, D_{i,s}, K, M, N, N_{\text{seg}}$
Initialize $x_{i,n}$, $R_{i,n} = 0 \,\forall\, i$, $n = 1, 2, \ldots N$
for all $y \in \{65, 70, \ldots, 95\}$ **do**
 Reset $x_{i,n} = 0$, $\text{BS}^{\text{air}}_{k,n} = 1 \,\forall\, i, k, n$.
 for all segments s **do**
  for all BSs k **do**
   Find set of priority users $\mathcal{P}_{k,s}$, and compute $r^{\mathcal{P}}_i$ as in (14). Sort $\mathcal{P}_{k,s}$ in descending order of $r^{\mathcal{P}}_i$.
   while $\mathcal{P}_{k,s} \neq \phi$ and $\sum_{n \in \mathcal{N}^s} \text{BS}^{\text{air}}_{k,n} > 0$ **do**
    for all users $i \in \mathcal{P}_{k,s}$ **do**
     Find slot n^* with the largest rate as in (16).
     Set $x_{i,n^*} = \hat{r}_{i,n^*} / r^{\mathcal{P}}_i$.
     Set $\text{BS}^{\text{air}}_{k,n^*} = \text{BS}^{\text{air}}_{k,n^*} - x_{i,n^*}$
    end for
    Recompute $R_{i,n}$, $\mathcal{P}_{k,s}$, and $r^{\mathcal{P}}_i$.
   end while
   for all slots $n \in \mathcal{N}^s$ **do**
    Find user i^* with the largest $\hat{r}_{i,n} \,\forall\, i \in \mathcal{U}_{k,n}$.
    If $\hat{r}_{i^*,n} > r^{y}_{i^*}$, then $x_{i^*,n} = x_{i^*,n} + \text{BS}^{\text{air}}_{k,n}$.
   end for
  end for
 end for
 Calculate $p_{k,n}$ using (2), where $\text{BS}^{\text{load}}_{k,n} = 1 - \text{BS}^{\text{air}}_{k,n}$.
 Calculate $P^{y}_{\text{Net}} = \sum_{k=1}^{K} \sum_{n=1}^{N} p_{k,n}$ for this iteration.
end for
Determine y^* that produces the minimum P^{y}_{Net}.
return $\mathbf{x}$, $\mathbf{p}$

2) Opportunistic Prebuffering: While the airtime minimization stage provides users with their *immediate* needs efficiently, it does not capitalize on granting users their *future* content in advance when their rates are high. Implementing such prebuffering results in reduced overall airtime since bulk transmissions are made opportunistically in short durations; and thereafter, users are not served. However, the following question remains: When is a good time to prebuffer content to a user? A simple rate threshold will not work well for cases where users have unequal rate distributions over $\mathcal{N}$. We therefore use the previously defined rate percentile $\hat{r}^{y\%}_{i,n}$ metric as it provides each user with an independent threshold, derived from its own rate statistics. This is applied as follows: For each slot $n \in \mathcal{N}^s$, we first find the user i^* with the largest rate, i.e.,

$$i^* = \arg\max_i \; \hat{r}_{i,n} \qquad \forall\, i \in \mathcal{U}_{k,n}. \tag{17}$$

This rate is then compared with the user's yth rate percentile at n, and if $\hat{r}_{i^*,n} > r^{y\%}_{i^*,n}$, the user is allocated the remaining BS airtime at that slot, and the user airtime is updated to $x_{i^*,n} = x_{i^*,n} + \text{BS}^{\text{air}}_{k,n}$. This completes the two steps of rate allocation performed $\forall\, n \in \mathcal{N}^s$. The procedure is then repeated by each BS, for each segment in sequence, as outlined in Algorithm 1. The BS power consumption matrix $\mathbf{p}$ is then calculated using (2), where $\text{BS}^{\text{load}}_{k,n} = 1 - \text{BS}^{\text{air}}_{k,n}$.

Setting $\mathbf{y}$*:* The value of y can affect the resulting power savings and is dependent on the current network load. At low load, a higher y will cause users to only prebuffer at close to peak rates. This is more efficient, provided users do not thereafter fall short of their needs and request airtime before encountering another "peak." On the other hand, when load is high, a lower value of y is preferred to allow users to prebuffer more frequently, even if at moderate rates. Although intermediate values $y \in [70, 80]$ provide a good tradeoff, the optimum value can be determined by iterating the procedure for different values and selecting the rate allocation $\mathbf{x}$ that gives the minimum power consumption.

Algorithm 2 Segment Quality Algorithm

Require: $x_{i,n}, \hat{r}_{i,n}, q_{\max}, M, N, N_{\text{seg}}, S$
Initialize $q_{i,s,1} = 1 \,\forall\, i, s$ [lowest quality level]
for all users i **do**
 for all segments s **do**
  Set current segment quality to the highest level $l^* = q_{\max}$, $q_{i,s,l} = 1$ for $l = l^*$, $q_{i,s,l} = 0 \,\forall\, l \neq l^*$.
  while $l^* \geq 0$ and (3) is violated for any s **do**
   Lower current segment quality, $l^* = l^* - 1$, $q_{i,s,l} = 1$ for $l = l^*$, $q_{i,s,l} = 0 \,\forall\, l \neq l^*$.
  end while
 end for
end for
return $\mathbf{q}$

B. Segment Quality Algorithm

After determining the rate-allocation matrix $\mathbf{x}$ as specified in Algorithm 1, the user segment quality levels are planned. The objective is to determine the segment quality plan that maximizes quality while providing smooth playback. The idea is to iterate over the segments in sequence and greedily maximize the current segment quality, while ensuring that the future segments can be streamed, at least, at the lowest quality level. On average, the quality levels will be equal to l_{req}. This is achieved as follows: All the segments are first initialized to a quality level of 1. Then, at the start of each segment, the quality level is set to $q_{\max}$, and a check is made to ensure that constraint (3) is satisfied for $s, s+1, \ldots S$. If this is not the case, the current segment quality is iteratively decremented until the constraints are met or the quality level is zero. The complete procedure is outlined in Algorithm 2.

Note that Algorithm 2 is applied to each user independently as the RA has already been determined. A practical property of the algorithm is that it ensures users experience the highest quality level as soon as possible and for the longest possible duration. This is not guaranteed by solving (12) or (13) since, when a mix of low- and high-quality segments are prebuffered, they can be ordered arbitrarily while remaining equivalently optimal. Therefore, Algorithm 2 can be also used to "postprocess" the optimal result of $\mathbf{x}$ in the MILP solutions to generate $\mathbf{q}$ solutions that favor "early" high-quality streaming.

C. BS On/Off Switching

To determine the BS on/off status, we simply search each BS for long "silent" transmission duration over the look-ahead window, where there is zero load. This is accomplished by the following simple procedure: 1) Determine the set of time slots $\mathcal{N}_{\mathrm{On}}$ where $p_{k,n} > P_0$, implying that the BS is on; and 2) then, determine the difference between the successive time slots in $\mathcal{N}_{\mathrm{On}}$. If this is larger than n_{off}, it means that no transmission occurred for a duration long enough to turn the BS off for that period. A value of zero is subtracted from the first element of $\mathcal{N}_{\mathrm{On}}$ to account for the possibility of switching the BS off before the first start, and the last element of $\mathcal{N}_{\mathrm{On}}$ is subtracted from N to check for a turn off possibility at the end.

This completes the multistage PGS solution, which we refer to as PGS-MinPower-Alg. For the case where BSs cannot switch to deep sleep, we do not apply the BS on/off stage, and only airtime is minimized. This solution is denoted by PGS-MinAir-Alg. Finally, when implementing the LP of (15), the algorithm will be denoted PGS-MinAir-AlgLP.

D. Computational Complexity

We first determine the complexity of each stage of the PGS multistage solution. The airtime minimization step of the rate allocation involves computing (14) and sorting set $\mathcal{P}_{k,n}$, which has time complexity of $O(MN + M\log M)$. Then, rate allocation over the N_{seg} slots takes $O(MN_{\mathrm{seg}})$ time, leading to overall complexity of $O(MN + M\log M + MN_{\mathrm{seg}})$ for this step. The subsequent prebuffering includes computing the future rate percentile stage and takes $O(N_{\mathrm{seg}}(M + N\log N))$ time. After accounting for S segments for each user, we arrive at overall complexity of $O(MN^2)$ for Algorithm 1. In the segment quality algorithm, the core step is to evaluate constraint (3), which has, as time complexity, $O(N + S)$ for a single user. This step is repeated at most $q_{\max}$ times when the constraint is violated, and repeated for each segment and each user. The resulting complexity order is $O(q_{\max}MS(N + S))$. In the worst case $S = N$, and $q_{\max}$ is typically less than six, which gives a worst-case runtime of $O(MN^2)$. As the BS on/off procedure presented earlier has complexity of $O(KN^2)$, this leads to overall runtime of $O((M + K)N^2)$.

VI. Numerical Results

Here, we present numerical results that demonstrate the potential energy savings achieved by exploiting rate predictions in the PGS framework. We also investigate the effects of prediction errors on the performance of the PGS schemes.

A. Simulation Setup

We consider two network setups. The first is a single cell with vehicles moving along a highway that crosses through the cell, and the second is a three-BS network, which is also along a highway, with an inter-BS distance of 1 km, as shown in Fig. 6. For realistic vehicular mobility, we use the SUMO traffic simulator [40] to generate mobility traces with a flow of one vehicle per second. Vehicles arrive in groups of ten vehicles each, separated by 60 s. This creates the effect of vehicle grouping observed on highways.

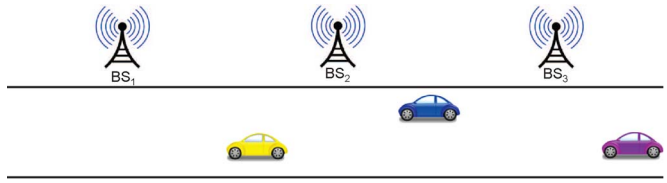

Fig. 6. Highway scenario with vehicular mobility.

BS transmit power is 40 W, and bandwidth is 5 MHz. BS power consumption at minimum and maximum loads is 200 and 1300 W, respectively, as presented for macro BSs employing time duty cycling in the power model of [14]. The minimum off time for a BS is set to 10 s. The slot duration is $\tau = 1$ s and $T = 240$ s. We consider a video format with four quality levels of $\{0.25, 0.5, 0.75, 1\}$ Mb/s, and a segment length $\tau_{\mathrm{seg}} = 10$ s. Gurobi 5.1 [41] was used to solve the PGS MILPs, and MATLAB was used as a simulation environment.

We compare the performance of the PGS schemes against two baseline approaches that do not exploit rate predictions. These reference schemes have two stages: rate allocation, followed by quality adaptation. Two rate-allocation schemes are considered: equal share (ES) and rate proportional (RP). In ES, airtime is shared equally among the users at each time slot. If there are $N_{k,n}$ users associated with BS k at time n, then $x_{i,n} = 1/N_{k,n}\ \forall\, i \in \mathcal{U}_{k,n}$, and the rate allocated to each is $\hat{r}_{i,n}/N_{k,n}$. The RP allocator is designed to be more spectrally efficient but not fair to users. Here, airtime is assigned to each user in proportion to its achievable data rate. Therefore, $x_{i,n} = \hat{r}_{i,n}/\sum_{i\in\mathcal{U}_{k,n}} \hat{r}_{i,n}$. Segment quality is then adapted based on the allocated rate at the start of the current segment, and the highest supportable level is selected. These approaches are referred to as ES-AdaptQ and RP-AdaptQ. We also consider a benchmark allocator that exploits rate predictions as in PGS. However, it is energy independent, and its objective is to maximize user quality. This is achieved by solving (13) with the objective of maximizing total video segment quality. This allocator serves as reference to what can be achieved with rate predictions, but without considering energy savings, and is referred to as MaxQuality-MILP.

The network-wide video quality and power consumption metrics are defined as follows.

- Q_{Net} is the total quality of all delivered segments divided by the number of requested segments.
- F_{Net} is the average percentage of playback time where the video is stalled over all users.
- P_{Net} is the average downlink power consumption of all BSs over the time window T.

B. Single-Cell Scenario

Fig. 7 shows the average BS power consumption versus the number of users M. As expected, the allocators consume more power with increasing M. The PGS-MinAir schemes achieve significant power savings by exploiting rate predictions, without having to power down the BSs. The energy gains can be also viewed as spectral efficiency gains, where the saved

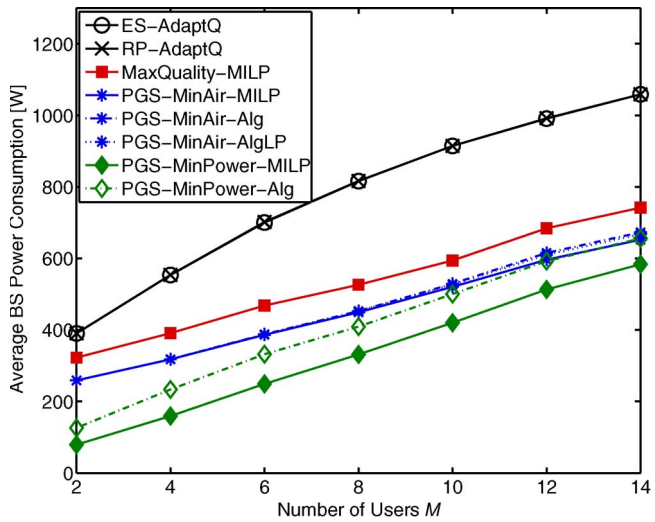

Fig. 7. BS downlink power consumption for varying number of users in the single-cell scenario.

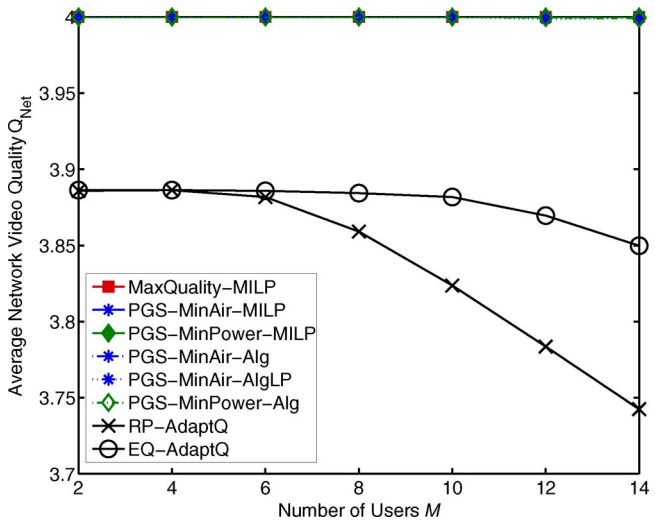

Fig. 8. Average quality level for varying number of users in the single-cell scenario.

airtime can be used for other users or applications. The MinAir-MILP and MinAir-AlgLP exhibit very close performance. This demonstrates the effectiveness of the developed multistage PGS framework that is able to achieve close to optimal performance with significant complexity reduction. Moreover, note that the MinAir-AlgLP and the MinAir-Alg achieve very close performance; therefore, the LP formulation of (15) can be replaced with the simple segment airtime minimization algorithm presented in Section V-A1, without performance loss. The MinPower-MILP scheme saves additional power by switching the BS to sleep intermittently and making bulk transmissions to users when awake. The sleep times are coordinated such that the users' QoS is not violated. When few users are present, the BS can sleep for prolonged periods; therefore, the power savings can be very large (approximately one eighth of the baseline allocator power is needed). However, as expected, for larger M, MinPower-MILP gradually converges to MinAir-MILP since, with many users, the BS cannot find sufficient time for a "sleep session." The MinPower-Alg performs close to the MinPower-MILP (exact solution) at low M, but it then deviates and converges to MinAir-Alg. The reason is that MinPower-MILP jointly optimizes the BS ON/OFF states with BS airtime minimization and is therefore able to strike the optimal tradeoff between serving users when their individual rates are high and grouping user transmissions together (even if not at their respective best rates) to generate blocks of sleep time. This, however, comes at the cost of a tightly coupled MILP that can take several minutes to solve.

In Fig. 8, we show the corresponding average segment quality level in this scenario. While the rate-predictive schemes all achieve the highest quality of four, the baseline schemes experience a slight quality degradation as M increases, with the RP-AdaptQ scheme suffering more. The video freezing, which is not depicted, was less than 1% for all allocators.

C. Multicell Highway Scenario

Fig. 9 shows average BS power consumption versus the number of users M for the three-BS network of Fig. 6. In this

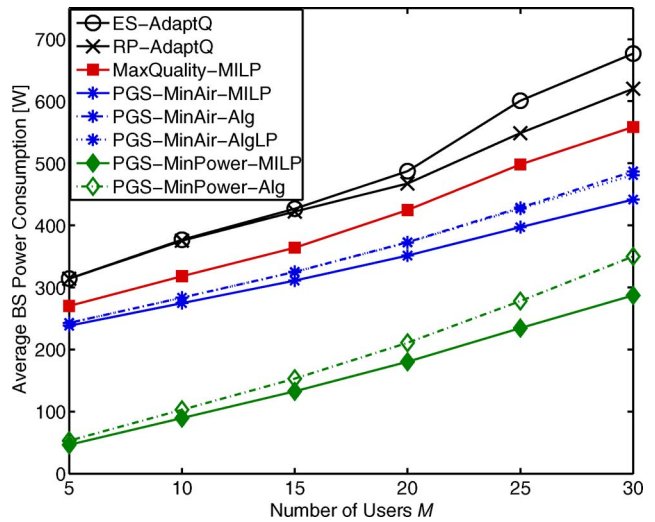

Fig. 9. BS downlink power consumption for varying number of users in the multicell highway scenario.

multicell scenario, the power-saving potential of the MinPower-MILP scheme is observed, while all the allocators achieve an average quality level of 3.75. User mobility information allows the BSs to sleep before users arrive in the cells. Further, as the allocation plan is made over three cells, a user may be granted all the video content in one or two of the BSs and nothing in the third (i.e., allowing it to sleep). In Fig. 9, we also note that the MinAir-Alg approaches deviate slightly from the MinAir-MILP solutions as M increases. With many users in a multicell network, the problem is more complex, and achieving optimality with the two-step rate-allocation algorithm is more difficult. A similar observation can be made for MinPower-Alg.

Fig. 10 shows the tradeoff that the PGS framework offers for video quality versus average BS power consumption. As shown, the power consumption of MinPower-MILP can be reduced by over 50% as the quality is decreased. The MinAir-MILP scheme also offers significant power reduction, albeit at a lower ratio. Note that the deviation of the multistage algorithm solutions from the MILP solutions increases with

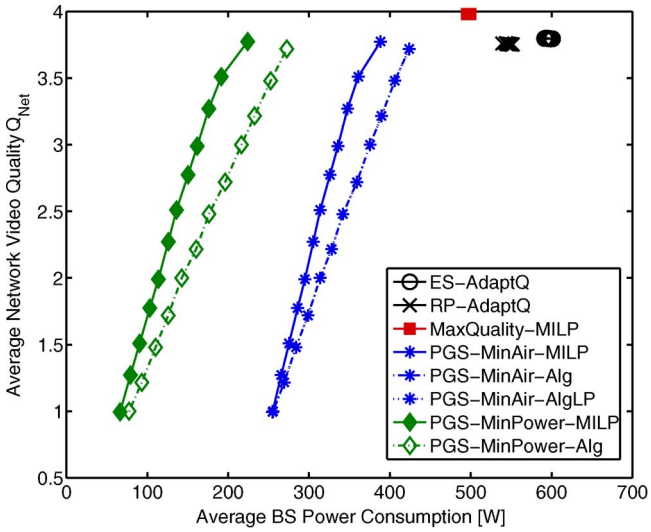

Fig. 10. Tradeoff between the average video quality and the BS power consumption in the multicell highway scenario.

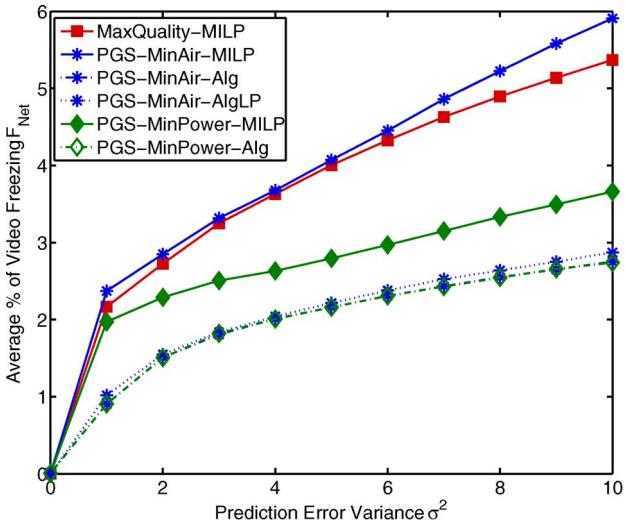

Fig. 11. Effect of prediction errors on video freezing of the PGS schemes in the highway scenario, $M = 20$.

quality, where again, a higher load makes its more difficult to achieve optimality.

D. Effect of Rate Prediction Errors

To evaluate the effect of rate prediction errors on the PGS schemes, we add a Gaussian random variable with a mean of zero and a variance σ^2 to the predicted user SNR. The resulting user rate matrix is denoted $\tilde{\mathbf{r}}$. Therefore, while the PGS schemes use $\hat{\mathbf{r}}$ to minimize power, the actual rates received are determined by $\mathbf{x} \odot \tilde{\mathbf{r}}$. This can degrade video quality and cause video freezing if the resulting allocation does not completely download the segments in their due time. Fig. 11 shows the impact of such errors on the video freezing for an increasing error variance σ^2. As expected, a higher error variance increases the video stalls. Interestingly, the algorithm-based PGS schemes are more robust to prediction errors, and achieve under 3% freezing for a high error variance. This indicates that even trends in the future user rates can provide significant power

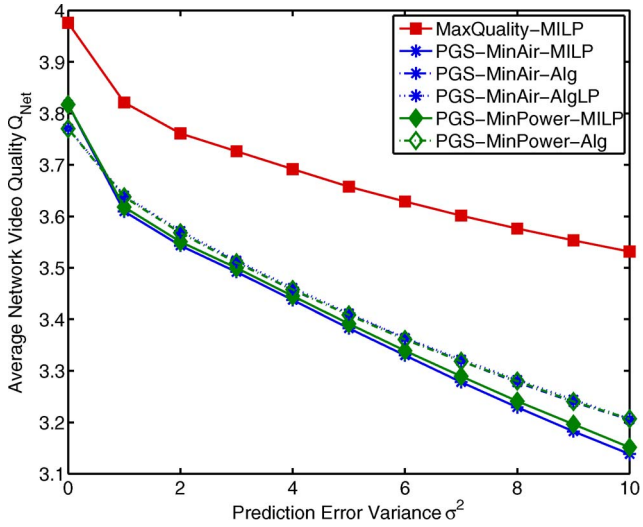

Fig. 12. Effect of prediction errors on the average video quality of the PGS schemes in the highway scenario, $M = 20$.

gains with minimal QoS loss. There are two main reasons behind the larger MILP solution sensitivity to prediction errors. First, since PGS-MILPs provide the lowest total airtime, when the observed rates are less than predicted, the user will be allocated an even lower rate, resulting in more freezing. This also explains why MinAir-MILP suffers more than MinPower-MILP (which has a larger airtime but lower power due to the sleep modes). Second, the optimization-based approaches make discrete allocation bursts, as shown in Fig. 4(b). While being optimal, these bursts can be spaced out in time (to wait for user channel peaks). Therefore, when the actual rate is less than predicted, the user has to wait until the next allocation to resume playback. In contrast, the PGS rate-allocation algorithm performs allocation every N_{seg} slots, when a user does not have any buffered segments. Fig. 12 also shows the effect of prediction errors on the average video quality and illustrates that the PGS schemes have more or less a similar quality sensitivity to errors, while the MaxQuality-MILP is more robust.

To investigate the effect of fast fading, we model the channel with i.i.d. Rayleigh-fading as well. The resultant $\tilde{\mathbf{r}}$ is now computed from an SNR that has a Gaussian error component and a fast-fading component. The results are shown in Figs. 13 and 14, where it is shown that, even with an error variance of zero, fast fading causes performance losses. Note that the relative effects of errors on the different PGS solutions follow similar trends to the previous results in Figs. 11 and 12. To improve the performance under effects of fast fading, we suggest that a more conservative measure of $\hat{\mathbf{r}}$ can be used while solving PGS. In other words, the values of $\hat{\mathbf{r}}$ can be decreased by a small factor to reduce the error effects on freezing, which happens when the actual rate is less than the predicted rate.

VII. Conclusion

In this paper, we have investigated how knowledge of future wireless data rates can improve spectral efficiency and provide downlink BS power savings. We used predicted rates to jointly optimize multiuser rate allocation, video segment quality, and

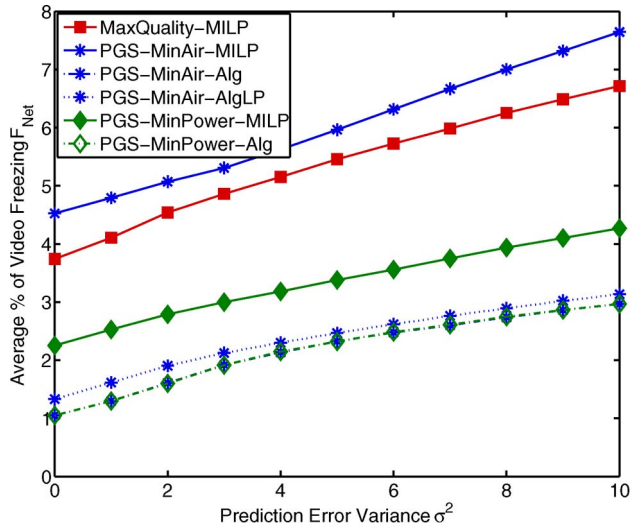

Fig. 13. Effect of prediction errors and fast fading on video freezing of the PGS schemes in the highway scenario, $M = 20$.

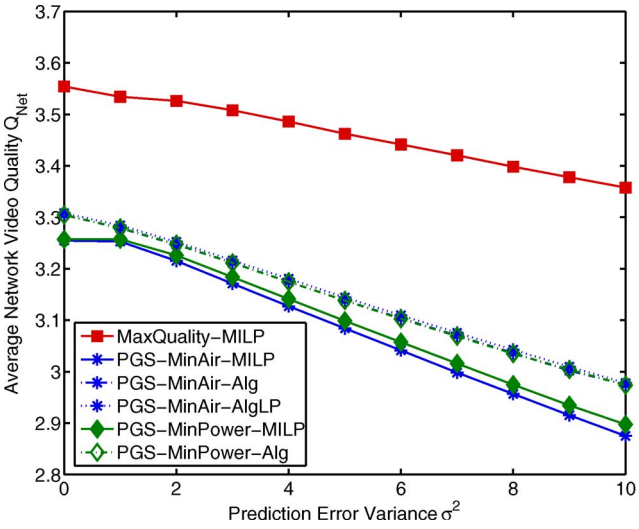

Fig. 14. Effect of prediction and fast fading on the average video quality of the PGS schemes in the highway scenario, $M = 20$.

BS on/off status. This was accomplished in an MILP formulation that captures the user video streaming requirements, the BS power consumption, and deep sleep mode operation. As the resulting MILP can be computationally intractable for large problem sizes, a multistage polynomial-time algorithm was developed. Simulations demonstrate that high energy efficiency gains are achieved by the proposed adaptive video transmission framework. Our numerical results indicate that the proposed PGS algorithms achieve close to optimal performance while exhibiting a higher degree of QoS robustness to prediction errors. Future work includes evaluating the potential of PGS to prolong the battery life of UEs as well. The fact that our PGS algorithm is less sensitive to prediction errors than the MILP formulation shows that there is room for further work. We thus plan to use stochastic channel models along with robust optimization techniques to improve the performance of PGS under uncertainty.

REFERENCES

[1] Z. Hasan, H. Boostanimehr, and V. Bhargava, "Green cellular networks: A survey, some research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 13, no. 4, pp. 524–540, 4th Quart., 2011.

[2] T. Chen, Y. Yang, H. Zhang, H. Kim, and K. Horneman, "Network energy saving technologies for green wireless access networks," *IEEE Wireless Commun.*, vol. 18, no. 5, pp. 30–38, Oct. 2011.

[3] *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update 2013_2018*, CISCO, San Jose, CA, USA, 2014.

[4] R. Pantos, W. May, and Apple Inc., HTTP Live Streaming. [Online]. Available: http://tools.ietf.org/html/draft-pantos-http-live-streaming-11, Apr. 2013, Accessed Jul. 27, 2013.

[5] *Dynamic Adaptive Streaming Over HTTP (DASH)—Part 1: Media Presentation Description and Segment Formats*, Int. Std. TS 26.247 V11.2.0, Apr. 2012, ISO/IEC.

[6] A. Begen, T. Akgul, and M. Baugher, "Watching video over the web: Part 1: Streaming protocols," *IEEE Internet Comput.*, vol. 15, no. 2, pp. 54–63, Mar./Apr. 2011.

[7] M. Malmirchegini and Y. Mostofi, "On the spatial predictability of communication channels," *IEEE Trans. Wireless Commun.*, vol. 11, no. 3, pp. 964–978, Mar. 2012.

[8] J. Yao, S. S. Kanhere, and M. Hassan, "An empirical study of bandwidth predictability in mobile computing," in *Proc. ACM Int. Workshop WiNTECH*, 2008, pp. 11–18.

[9] J. Johansson, W. Hapsari, S. Kelley, and G. Bodog, "Minimization of drive tests in 3GPP," *IEEE Commun. Mag.*, vol. 50, no. 11, pp. 36–43, Nov. 2012.

[10] X. Chen, F. Mériaux, and S. Valentin, "Predicting a user's next cell with supervised learning based on channel states," in *Proc. IEEE Int. Workshop SPAWC*, Jun. 2013, pp. 36–40.

[11] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi, "Understanding individual human mobility patterns," *Nature*, vol. 453, no. 7196, pp. 779–782, Jun. 2008.

[12] C. Song, Z. Qu, N. Blumm, and A. Barabasi, "Limits of predictability in human mobility," *Science*, vol. 327, no. 5968, pp. 1018–1021, Feb. 2010.

[13] S. Deng and H. Balakrishnan, "Traffic-aware techniques to reduce 3G/LTE wireless energy consumption," in *Proc. 8th Int. CoNEXT*, 2012, pp. 181–192.

[14] C. Desset, B. Debaillie, V. Giannini, A. Fehske, G. Auer, and H. Holktamp, "Flexible power modeling of LTE base stations," in *Proc. IEEE WCNC*, Apr. 2012, pp. 2858–2862.

[15] L. Correia, D. Zeller, O. Blume, D. Ferling, A. Kangas, I. Godor, G. Auer, and L. Van der Perre, "Challenges and enabling technologies for energy aware mobile radio networks," *IEEE Commun. Mag.*, vol. 48, no. 11, pp. 66–72, Nov. 2010.

[16] I. Ashraf, F. Boccardi, and L. Ho, "Sleep mode techniques for small cell deployments," *IEEE Commun. Mag.*, vol. 49, no. 8, pp. 72–79, Aug. 2011.

[17] M. Ismail and W. Zhuang, "Network cooperation for energy saving in green radio communications," *IEEE Wireless Commun.*, vol. 18, no. 5, pp. 76–81, Oct. 2011.

[18] F. Han, Z. Safar, W. Lin, Y. Chen, and K. Liu, "Energy-efficient cellular network operation via base station cooperation," in *Proc. IEEE ICC*, 2012, pp. 4374–4378.

[19] T. Han and N. Ansari, "On greening cellular networks via multicell cooperation," *IEEE Wireless Commun.*, vol. 20, no. 1, pp. 82–89, Feb. 2013.

[20] S. H. Ali, V. Krishnamurthy, and V. C. M. Leung, "Optimal and approximate mobility-assisted opportunistic scheduling in cellular networks," *IEEE Trans. Mobile Comput.*, vol. 6, no. 6, pp. 633–648, Jun. 2007.

[21] H. Abou-zeid, H. Hassanein, and S. Valentin, "Optimal predictive resource allocation: Exploiting mobility patterns and radio maps," in *Proc. IEEE GLOBECOM*, 2013, pp. 4714–4719.

[22] H. Abou-zeid, H. S. Hassanein, and N. Zorba, "Long-term fairness in multi-cell networks using rate predictions," in *Proc. IEEE GCC Conf. Exhib.*, 2013, pp. 131–135.

[23] R. Margolies, A. Sridharan, V. Aggarwal, R. Jana, N. K. Shankaranarayanan, V. A. Vaishampayan, and G. Zussman, "Exploiting mobility in proportional fair cellular scheduling: Measurements and algorithms," in *Proc. IEEE INFOCOM*, to be published.

[24] J. Yao, S. Kanhere, and M. Hassan, "Improving QoS in high-speed mobility using bandwidth maps," *IEEE Trans. Mobile Comput.*, vol. 11, no. 4, pp. 603–617, Apr. 2012.

[25] I. D. Curcio, V. K. M. Vadakital, and M. M. Hannuksela, "Geo-predictive real-time media delivery in mobile environment," in *Proc. ACM Workshop MoViD*, 2010, pp. 3–8.

[26] H. Riiser, T. Endestad, P. Vigmostad, C. Griwodz, and P. Halvorsen, "Video streaming using a location-based bandwidth-lookup service for bitrate planning," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 8, no. 3, pp. 24:1–24:19, Jul. 2012.

[27] J. Hao, R. Zimmermann, and H. Ma, "GTube: Geo-predictive video streaming over HTTP in mobile environment," in *Proc. ACM MMSYS Conf.*, 2014, pp. 259–270.
[28] H. Abou-zeid, H. S. Hassanein, and N. Zorba, "Enhancing mobile video streaming by lookahead rate allocation in wireless networks," in *Proc. IEEE CCNC*, 2014, pp. 768–773.
[29] J. Chen, R. Mahindra, M. A. Khojastepour, S. Rangarajan, and M. Chiang, "A scheduling framework for adaptive video delivery over cellular networks," in *Proc. ACM Int. Conf. Mobile Comput. Netw.*, 2013, pp. 389–400.
[30] D. De Vleeschauwer, H. Viswanathan, A. Beck, S. Benno, G. Li, and R. Miller, "Optimization of HTTP adaptive streaming over mobile cellular networks," in *Proc. IEEE INFOCOM*, 2013, pp. 898–997.
[31] V. Joseph and G. de Veciana, "NOVA: QoE-driven optimization of dash-based video delivery in networks," in *Proc. IEEE INFOCOM*, to be published.
[32] Z. Lu and G. de Veciana, "Optimizing stored video delivery for mobile networks: The value of knowing the future," in *Proc. IEEE INFOCOM*, 2013, pp. 2706–2714.
[33] H. Abou-zeid and H. S. Hassanein, "Predictive green wireless access: Exploiting mobility and application information," *IEEE Wireless Commun.*, vol. 20, no. 5, pp. 92–99, Oct. 2013.
[34] H. Abou-zeid and H. S. Hassanein, "Efficient lookahead resource allocation for stored video delivery in multi-cell networks," in *Proc. IEEE WCNC*, pp. 1932–1937.
[35] "LTE; E-UTRA; Radio frequency system scenarios," Sophia-Antipolis, France, Tech. Rep. TR 36.942 V11.0.0, 3GPP, Sep. 2012.
[36] O. Oyman and S. Singh, "Quality of experience for HTTP adaptive streaming services," *IEEE Commun. Mag.*, vol. 50, no. 4, pp. 20–27, Apr. 2012.
[37] "Energy efficiency analysis of the reference systems, areas of improvements and target breakdown," Brussels, Belgium, Tech. Rep. INFSO-ICT-247733, Deliverable D2.3, EARTH, Jan. 2012.
[38] *Coding of Moving Pictures and Audio, 2013*, ISO/IEC JTC1/SC29/WG11, 2013, Accessed Feb. 20, 2014.
[39] M. Bazaraa, J. Jarvis, and H. Sherali, *Linear Programming and Network Flows*, 3rd ed. Hoboken, NJ, USA: Wiley, 2005.
[40] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "SUMO-Simulation of urban MObility: An overview," in *Proc. 3rd Int. Conf. Adv. Syst. SIMUL*, 2011, pp. 63–68.
[41] Gurobi, Houston TX, USA, Gurobi Optimization, Accessed Feb. 11, 2014. [Online]. Available: http://www.gurobi.com/.

Hatem Abou-zeid (S'04) received the B.Sc. (Hons.) and M.Sc. (Hons.) degrees in communication engineering from the Arab Academy for Science, Technology and Maritime Transport, Alexandria, Egypt, in 2005 and 2008, respectively. He is currently working toward the Ph.D. degree with Queen's University, Kingston, ON, Canada.

He is also currently a Research Assistant with Queen's University. He is an experienced Lecturer and has been granted several Teaching Fellowships at Queen's University to instruct freshman and senior-level engineering courses. His research interests include context-aware radio access networks, network adaptation and cross-layer optimization, adaptive video delivery, and vehicular communications.

Mr. Hatem is a Technical Reviewer for several prestigious conferences and journals. He received a DAAD RISE-Pro Research Scholarship in 2011 for a six-month internship at Bell Labs, Germany.

Hossam S. Hassanein (S'86–M'90–SM'06) received the B.Sc. (Hons.) degree in electrical engineering from Kuwait University, Kuwait City, Kuwait, in 1984, the M.Sc. degree in computer engineering from the University of Toronto, Toronto, ON, Canada, in 1986, and the Ph.D. degree in computer science from the University of Alberta, Edmonton AB, Canada, in 1990.

He is the Founder and Director of the Telecommunications Research Laboratory, School of Computing, Queen's University, Kingston, ON, Canada, with extensive international academic and industrial collaborations. He is a leading authority in the areas of broadband, wireless, and mobile network architecture, protocols, control, and performance evaluation. His record spans more than 500 publications in journals, conferences, and book chapters, in addition to numerous keynotes and plenary talks at flagship venues.

Dr. Hassanein served as the Chair for the IEEE Communication Society Technical Committee on Ad hoc and Sensor Networks. He is an IEEE Communications Society Distinguished Speaker (Distinguished Lecturer 2008–2010). He has received several recognitions and Best Paper awards at top international conferences.

Stefan Valentin (S'07–M'10) received the M.A. (with excellence) degree in communication and electrical engineering from the Technical University Berlin, Germany, in 2006 and the Ph.D. (*summa cum laude*) degree in computer science from the University of Paderborn, Germany, in 2010.

Since 2010, he has been a Full Researcher with the Alcatel-Lucent Bell Labs, Stuttgart, Germany. Previous appointments include the International Centre of Theoretical Physics, Trieste, Italy; and Carleton University, Ottawa, ON, Canada. His main research interests are cooperative relaying, wireless resource allocation, and context-aware radio access.

Dr. Valentin has advised the German Federal Ministry of Education and Research since 2013, has been a member of the Alcatel-Lucent Leadership Program since 2012, and has led the Ph.D. Internship Program at Bell Labs, Germany, since 2010. He received the SIMUTools Best Paper Award in 2008, the Klaus Tschira Award for Comprehensible Science in 2011, and the Bell Labs Special Award of Excellence in 2013.