Computer Communications 35 (2012) 1828–1837

Contents lists available at SciVerse ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

A majorization approach to downlink multiuser VBR video streaming

Yingsong Huang, Shiwen Mao [*], Yihan Li

Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849-5201, USA

ARTICLE INFO

Article history:
Available online 28 January 2012

Keywords:
Majorization
Power control
Stochastic programming
Variable bit rate
Video streaming

ABSTRACT

In this paper, we investigate the problem of optimal power control for multiuser variable bit rate (VBR) video streaming in a cellular network with orthogonal channels. We adopt a deterministic model for VBR video traffic that incorporates video frame and playout buffer characteristics, and formulate a constrained stochastic optimization problem. We then develop a majorization-based solution approach. For the case of a single VBR video session with relaxed peak power constraint, we develop a power optimal algorithm with low complexity. We prove the power optimality of the proposed algorithm and the uniqueness of the global optimum, and demonstrate that the proposed algorithm is also smoothness optimal. For the case of multiuser VBR video streaming, we develop a heuristic algorithm that selectively suspends some video sessions when the peak power constraint is violated. In addition to the traditional VBR video streaming application, we also consider the case of interactive video streaming, and show that the proposed schemes can be easily adapted and applied. The proposed algorithms are evaluated with trace-driven simulations, and are shown to achieve considerable power savings and improved video quality over a conventional "lazy" scheme.

1. Introduction

With the wide availability of intelligent handheld devices and smart phones, there is a compelling need to support video in traditional cellular networks. Such applications not only are bandwidth intensive, but also involve user quality of experiences. They will greatly stress the capacity of traditional cellular networks and call for joint design and optimization of networking and control mechanisms in multiple layers.

Among various forms of compressed videos, *variable bit rate* (VBR) videos can offer constant and better quality over constant bit rate (CBR) videos with the same bit budget; a highly desirable advantage for video applications [1]. However, VBR videos also pose great challenges to network scheduling and control, due to the high variability and complex autocorrelation structure in VBR video traffic [2].

In our prior work [1], we investigate the problem of supporting VBR videos in a multi-cell network, where capacity is limited by inter-cell interference. We develop a reformulation–linearization technique (RLT) approach as well as a distributed algorithm based on dual decomposition. In this paper, we focus on the downlink of a cell with orthogonal channels, where the base station (BS) streams multiple VBR videos to mobile users in the cell. We consider buffer underflows (causing stalled display) and overflows (causing missing frames and error propagations in the following received and decoded frames) as user viewing performance measure, and aim to minimize the total power consumption at the BS.

We adopt a deterministic model for VBR video traffic that incorporates video frame and playout buffer characteristics, and formulate a *constrained stochastic optimization* problem. We show that the problem fits well with *majorization theory*, which concerns with partial ordering of real vectors and order-preserving functions, and develop a majorization-based solution framework. First, we prove that the objective function of the formulated problem is *Schur-convex* with the order-preserving property [3]. Secondly, we investigate the case of a single VBR video session with relaxed peak power constraint. We develop a majorization-based power optimal algorithm with low complexity. We prove the *power optimality* of the proposed algorithm and the *uniqueness* of the global optimum, and demonstrate that the proposed algorithm is also *smoothness optimal*. Thirdly, we investigate the case of multiuser VBR streaming, where power allocations for the users are coupled with the peak power constraint. We develop a heuristic algorithm that selectively suspends some video sessions, which will not incur underflow in the next time slot, when the peak power constraint is violated.

In addition to the traditional video streaming application, we also consider how to support interactive video streaming, a relative new problem area with a broad scope [4–6]. We focus on the three typical scenarios in this paper: (i) interactive video streaming with tight delay constraints, which translates to a limited amount information of future frame sizes and small playout buffer sizes; (ii)

[*] Corresponding author. Tel.: +1 334 844 1845; fax: +1 334 844 1809.
E-mail addresses: yzh0002@tigermail.auburn.edu (Y. Huang), smao@ieee.org (S. Mao), yli@auburn.edu (Y. Li).

0140-3664/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2012.01.020

Table 1
Notation.

Symbol	*Description*
N	Number of mobile users in the cell
$\mathcal{U}$	Set of users
b_n	Playout buffer size at user n
T_n	Total number of frames of the user n video
Ω_n	Total number of bits of the use n video
$D_n(t)$	Cumulative consumption curve of user n
$B_n(t)$	Cumulative overflow curve of user n
$X_n(t)$	Cumulative transmission curve of user n
$P_n(t)$	Transmit power of user n in time slot t
$\vec{P}(t)$	Power allocation in time slot t
$\gamma_n(t)$	SINR at user n in time slot t
$G_n(t)$	Path gain from BS to user n in time slot t
$\eta_n(t)$	Noise power at user n in time slot t
$c_n(t)$	Downlink data rate of user n in time slot t
B_w	Channel bandwdith
κ	A transceiver dependent constant
$\overline{P}$	Peak power constraint
$\vec{C}_n^i$	The i-th feasible transmission schedule
$\vec{C}_n^*$	The optimal solution to (10)
$\vec{C}_n^{opt}$	An evenly distributed rate vector
$\vec{C}_n^1$	An auxiliary schedule used in Theorem 2 proof
$\Phi(\cdot)$	A mapping function $\mathcal{R}^{T_n} \to \mathcal{R}$ defined in (13)
$\vec{X}, \vec{Y}, \vec{Z}$	n-dimensional nonnegative vectors
$C_{max}(t), C_{min}(t)$	Rate of probe lines
$U(\vec{C})$	Smoothness utility function

streaming interactive videos with VCR controls [7], where some frames are skipped or replayed during the streaming process; and (iii) the new types of "*exploratory*" online interactive videos and *video click throughs* [8]. We show that the proposed schemes can be easily adapted for the three typical scenarios to achieve both power savings and improved video quality over conventional schemes.

The proposed algorithms are evaluated with trace-driven simulations for both conventional stored VBR video streaming and interactive video streaming. They are shown to achieve considerable power savings over a conventional "lazy" scheme. This finding is somewhat *counter-intuitive*, since the lazy scheme seems to be energy efficient by only transmitting the minimal amount of video data in each time slot.

The rest of this paper is organized as follows. The system model and problem statement are presented in Section 2. We transform the problem into a majorization problem in Section 3. The proposed algorithms are described in Section 4 and simulation results are presented in Section 5. We review related work in Section 6. Section 7 concludes this paper. The notation used in this paper is summarized in Table 1.

2. System model and problem statement

2.1. Network and video system model

We consider the downlink of a cellular network, as shown in Fig. 1. There are N mobile users in the cell. Let $\mathcal{U} = \{1, 2, \ldots, N\}$ denote the set of users. A BS streams multiple VBR videos to the mobile users. Each user occupies a downlink channel, which is a spectral/time resource slot, the nature of which depends on the specific multiple access technique adopted. We assume that the downlink channels within a cell are orthogonal, due to perfect synchronization of the spreading codes or the use of guard times or frequencies.

We assume the wireline segment of a video session path is reliable with sufficient bandwidth, while the last-hop wireless link is the bottleneck. Thus the corresponding video data is always available at the BS before the scheduled transmission time. It is nontrivial to accurately model VBR video traffic, which exhibits both strong asymptotic self-similarity and short-range correlation [2]. To this end, we adopt a *deterministic model* that considers frame sizes, frame intervals, and playout buffers [9]. Let $D_n(t)$ denote the *cumulative consumption curve* of the n-th user, representing the cumulative amount of bits consumed by the decoder at time t. The cumulative consumption curve is determined by video characteristics such as frame sizes and frame rates. User n has a playout buffer of b_n bits. Its video has T_n, frames, which translates to Ω_n bits in total. We can derive a *cumulative overflow curve* for user n as

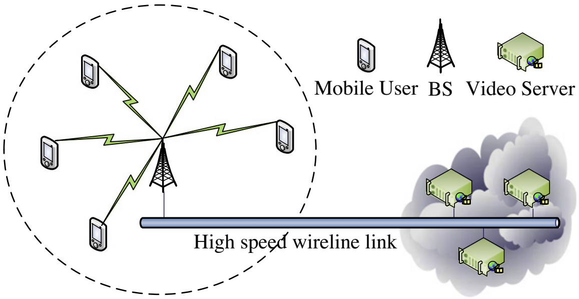


Fig. 1. Downlink multiuser video streaming system model.

$$B_n(t) = \min\{D_n(t-1) + b_n, D_n(T_n)\}, \quad 0 \leqslant t \leqslant T_n. \tag{1}$$

$B_n(t)$ is the maximum number of received bits at time t without overflowing user n's playout buffer. Finally we define *cumulative transmission curve* $X_n(t)$ as the total amount of bits transmitted to user n at time t.

The three curves are illustrated in Fig. 2. We assume time is divided into time slots, and power control is performed on time slot units. Note that the frame intervals of the VBR videos do not have to be aligned, and time slots do not have to be aligned with the frame intervals. To simplify notation, we assume the frames are aligned and the duration of a time slot is τ in the rest of this paper. A feasible transmission schedule will produce a cumulative transmission curve $X_n(t)$ that lies in between $D_n(t)$ and $B_n(t)$, i.e., causing neither underflow nor overflow at the playout buffer. In practice, $D_n(t)$'s are known for stored videos and can be delivered to the BS as metadata during the session setup phase, and $B_n(t)$'s are then derived as in (1) based on playout buffer size information.

The BS allocates a transmit power to each user in each time slot. Let $\vec{P}(t) = [P_1(t), \ldots, P_N(t)]$ be the power allocation in time slot t. The Signal to Interference-plus-Noise Ratio (SINR) at user n in time slot t can be written as

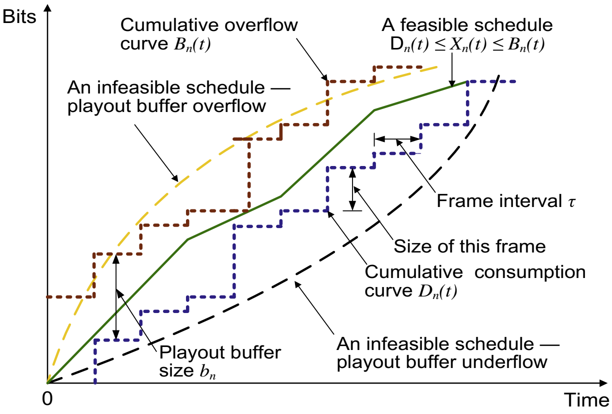


Fig. 2. Transmission schedules for VBR video session n.

$$\gamma_n(t) = G_n(t)P_n(t)/\eta_n(t), \tag{2}$$

where $G_n(t)$ is the path gain from BS to user n and $\eta_n(t)$ is the the noise power at user n in time slot t. We assume block fading channels, where the $G_n(t)$'s are i.i.d. random variables with a certain distribution, for $t = 1, \dots, T_n$ [10]. The downlink data rate can be written as $c_n(t) = B_w \log(1 + \kappa\gamma_n(P_n(t)))$, where B_w is the channel bandwidth and κ depends on the transceiver design, such as modulation and channel coding. Without loss of generality, we use the Shannon capacity as an upper bounding approximation:

$$c_n(t) = B_w \log(1 + \gamma_n(P_n(t))). \tag{3}$$

Once the link capacity is determined, $c_n(t)\tau$ bits of video data will be delivered to user n in that time slot. The cumulative transmission curve $X_n(t)$ can be written as

$$X_n(0) = 0; \quad X_n(t) = X_n(t-1) + c_n(t)\tau. \tag{4}$$

A feasible transmission schedule should cause neither playout buffer underflow nor overflow, i.e., satisfying

$$D_n(t) \leqslant X_n(t) \leqslant B_n(t), \quad \text{for all } t, n. \tag{5}$$

2.2. Problem statement

The problem is to find the optimal feasible transmission schedules $\{X_n(t), 0 < t \leqslant T_n\}$, for all user $n \in \mathcal{U}$, such that the total transmit power consumption can be minimized. From (3), the required transmit power for user n is

$$P_n(t) = (2^{c_n(t)/B_w} - 1)\eta_n(t)/G_n(t). \tag{6}$$

A peak power constraint may be applied at the BS, i.e., $\sum_{n\in\mathcal{U}} P_n(t) \leqslant \overline{P}$, for all t. We then formulate the following *constrained stochastic optimization problem*, to minimize the expectation of the total transmit power.

$$\text{minimize} \quad \sum_{n\in\mathcal{U}} \sum_{t=1}^{T_n} \mathbb{E}[P_n(t)] \tag{7}$$

$$\text{subject to}: \; D_n(t) \leqslant X_n(t) \leqslant B_n(t), \quad \text{for all } n, t \tag{8}$$

$$\sum_{n\in\mathcal{U}} P_n(t) \leqslant \overline{P}, \quad \text{for all } t. \tag{9}$$

Due to orthogonal channels, transmission in one channel does not interfere with those in other channels. We first relax the peak power constraint (9) (i.e., the case when $\overline{P}$ is large). Then, Problem (7) can be decomposed into N sub-problems, each minimizing the transmit power of a video session.

$$\text{minimize} \quad \sum_{t=1}^{T_n} \mathbb{E}[P_n(t)], \quad \text{for all } n \in \mathcal{U} \tag{10}$$

$$\text{subject to}: \quad D_n(t) \leqslant X_n(t) \leqslant B_n(t), \quad \text{for all } n, t. \tag{11}$$

For given $B_n(t)$ and $D_n(t)$, the feasible transmission schedule satisfying (11) is not unique. The i-th feasible transmission schedule is a piece-wise linear curve that can be represented as a vector $\vec{C}_n^i = [c_n^i(1), \dots, c_n^i(T_n)]$, where $c_n^i(t) \geqslant 0$ is the data rate in time slot t, for all t. Let $\vec{C}_n^* = [c_n^*(1), \dots, c_n^*(T_n)]$ be the optimal solution to (10). For a given VBR video, all the feasible transmission schedules transmit the same amount of video data, i.e.,

$$\sum_{t=1}^{T_n} c_n^i(t) = \sum_{t=1}^{T_n} c_n^*(t) = \Omega_n, \quad \text{for all } i, n. \tag{12}$$

Furthermore, the total transmit power for a feasible schedule can be viewed as a mapping function $\Phi : \mathcal{R}^{T_n} \to \mathcal{R}$ with

$$\Phi(\vec{C}_n^i) = \sum_{t=1}^{T_n} (2^{c_n^i(t)/B_w} - 1)\eta_n(t)/G_n(t). \tag{13}$$

Given such an interpretation of Problem (10), the objective is to find an optimal feasible vector $\vec{C}_n^*$, such that its total power P_n^*, obtained through the mapping $\Phi(\cdot)$, is the minimum among all feasible vectors $\vec{C}_n^i$. This interpretation fits well with the *majorization* theory, which provides useful order preserving results for inequality problems [3]. Applying these results, we design an optimal algorithm for solving the decomposed sub-problem (10) in Section 4.1. We will examine then the case of multiuser VBR video streaming coupled with the peak power constraint in Section 4.2.

3. Inequality and theory of majorization

3.1. Majorization preliminaries

Majorization theory formalizes the intuitive notion of inequality and describes when the components of a vector are "less spread out" or "more nearly equal" than the components of another vector [3]. It concerns with how to order vectors with nonnegative real components and order-preserving functions. Majorization theory has been used to solve communications and networking problems in the literature [9,11,12]. In this paper, we apply majorization to the problem of optimal power control for streaming VBR videos. For simplicity, all the vectors in this section are row vectors.

Definition 1. Consider two n-dimensional vectors $\vec{X}$ and $\vec{Y}$ with nonnegative real elements. Let the elements be ordered non-increasingly and re-indexed as $x_1 \geqslant \cdots \geqslant x_n \geqslant 0$ for $\vec{X}$ and $y_1 \geqslant \cdots \geqslant y_n \geqslant 0$ for $\vec{Y}$. $\vec{X}$ is said to be majorized by $\vec{Y}$, denoted as $\vec{X} \prec \vec{Y}$, if $\sum_{i=1}^t x_i \leqslant \sum_{i=1}^t y_i, t = 1, \dots, n-1$ and $\sum_{i=1}^n x_i = \sum_{i=1}^n y_i$ [3].

For example, we have $[1/n, \dots, 1/n] \prec [1/(n-1), \dots, 1/(n-1), 0] \prec [1/2, 1/2, 0, \dots, 0] \prec [1, 0, \dots, 0]$. It can be seen that the total weight of 1 becomes more and more concentrated into fewer and fewer elements in the above majorized vector sequence. When such partial ordering is obtained, an *order-preserving function* can be applied to the vectors and the outcomes remain monotonic with respect to the partial order of the vectors.

Definition 2. A real-valued function $\phi(\cdot)$ defined on a set $\mathcal{S} \subset \mathcal{R}^n$ is said to be *Schur-convex* on $\mathcal{S}$, if $\vec{X} \prec \vec{Y}$ implies $\phi(\vec{X}) \leqslant \phi(\vec{Y})$, for all $\vec{X}, \vec{Y} \in \mathcal{S}$ [3].

Schur-convex functions have the order-preserving property, which makes majorization useful for solving optimization problems. The following fact can be used to verify whether a function is Schur-convex.

Fact 1. *If $\phi(\cdot)$ is symmetric and convex, then ϕ is Schur-convex. Consequently, $\vec{X} \prec \vec{Y}$ implies $\phi(\vec{X}) \leqslant \phi(\vec{Y})$* [3].

If we choose $\phi = \sum_i f(x_i)$ and $f(\cdot)$ is continuous and convex, then we have the following strong fact.

Fact 2. *$\sum_i f(x_i) \leqslant \sum_i f(y_i) \Longleftrightarrow \vec{X} \prec \vec{Y}$ holds true for all continuous convex functions $f : \mathcal{R} \to \mathcal{R}$* [3].

We can extend the concept of majorization to the case of a set of vectors or multiple segments of a long vector.

Lemma 1. *Consider $\vec{X} = [\vec{X}_1, \dots, \vec{X}_K]$ and $\vec{Y} = [\vec{Y}_1, \dots, \vec{Y}_K]$, where each element vector $\vec{X}_i$ and $\vec{Y}_i$ has dimension J_i and satisfying $\vec{X}_i \prec \vec{Y}_i, i = 1, \dots, K$. Then we have $\vec{X} \prec \vec{Y}$.*

Proof. Let $f(\cdot)$ be a continuous and convex function. From Fact 2, we have $\vec{X}_i \prec \vec{Y}_i \Longleftrightarrow \sum_{j=1}^{J_i} f(x_i^j) \leqslant \sum_{j=1}^{J_i} f(y_i^j)$. It follows that $\sum_{i=1}^K \sum_{j=1}^{J_i} f(x_i^j) \leqslant \sum_{i=1}^K \sum_{j=1}^{J_i} f(y_i^j)$, i.e., $\vec{X} \prec \vec{Y}$. □

Consider three real vectors $\vec{X} = [\bar{x}, \dots, \bar{x}]$, $\vec{Y} = [y_1, \dots, y_n]$, and $\vec{Z} = [z_1, \dots, z_n]$, where $\sum_{i=1}^n y_i = \sum_{i=1}^n z_i = n\bar{x}$. If the elements in each vector are non-increasing, we can plot the cumulative curves for the normalized vectors $\vec{X}/(n\bar{x}), \vec{Y}/(n\bar{x})$ and $\vec{Z}/(n\bar{x})$, which are piece-wise linear curves A, B, and C in Fig. 3. The i-th point on the cumulative curve is the sum of the first i elements of the corresponding vector. If each element of the vectors is the income of an individual, these are known as *Lorenz Curves* that evaluate social income inequality [13]; each point $(a\%, b\%)$ on the curve indicates the top a% of the population earn b% of the income.

In Fig. 3, the straight line A is for vector $\vec{X}/(n\bar{x})$, i.e., equal distribution. Unequal distributions $\vec{Y}/(n\bar{x})$ and $\vec{Z}/(n\bar{x})$ produce curves B and C, respectively, which are bent in the middle. As the bow-shaped curves are bent, concentration increases. We call B and C *concave bow curves*. The bow curve closer to A represents more even distribution [3], which leads to $\vec{X} \prec \vec{Y} \prec \vec{Z}$. Furthermore, we can permute $\vec{Y}$ and $\vec{Z}$ by reordering their elements, to obtain new vectors $\vec{Y}'$ and $\vec{Z}'$. We call the resulting curves B' and C' *convex bow curves*. Since the order of elements in a vector plays no role in majorization (see Definition 1), $\vec{X} \prec \vec{Y}' \prec \vec{Z}'$ still holds true in this convex bow curve case.

3.2. Schur-convexity of Problem (10)

As discussed, Problem (10) fits well with majorization theory with a mapping function (13). To solve the problem, we need to find the optimal rate vector $\vec{C}_n^*$ that is majorized by all other feasible transmission rate vectors $\vec{C}_n^i$, as $\vec{C}_n^* \prec \vec{C}_n^i$, for all i. If the mapping (13) is Schur-convex, then the total transmit power to achieve $\vec{C}_n^*$ will also be dominated by those of other feasible transmission rate vectors. That is, the minimum power is found for Problem (10). Due to random path gains and noise powers, stochastic majorization (rather than ordinary majorization) should be used, which investigates the inequality properties related to random variables [3]. We have the following theorem for the mapping (13) in Problem (10).

Theorem 1. *The objective function of (10) is an increasing Schur-convex function.*

Proof. Due to i.i.d. channel gains and noise powers, the random variables $\eta_n(t)/G_n(t)$'s are *exchangeable*, for all t. Define $w(g, \eta, c) = (2^{c/B_w} - 1)\eta/g$, which is convex and increasing with c, for all $g \geqslant 0$ and $\eta \geqslant 0$. Let $\psi(\vec{C}) = \mathbb{E}[\Phi(\vec{C})] = \mathbb{E}[\sum_{t=1}^{T} w(g(t), \eta(t), c(t))]$. $\Phi(\vec{C})$ is a symmetric, convex and increasing function in $\vec{C}$ for each fixed $\vec{G}$ and $\vec{\eta}$. According to Proposition 11.B.5 in [3], $\psi(\vec{C})$ is symmetric, convex and increasing. Following Fact 1, the objective function (10) is Schur-convex and increasing. □

With Theorem 1, solving Problem (10) is equivalent to finding the optimal rate vector $\vec{C}_n^*$, such that $\vec{C}_n^* \prec \vec{C}_n^i$, for all i. Then the total power associated with $\vec{C}_n^*$ is the minimum since the mapping (13) is order-conserving. As discussed in Section 3.1, the feasible rate vector that is closest to equal distribution (i.e., curve A in Fig. 3) will be majorized by all other feasible rate vectors. Therefore, we transform Problem (10) to find a transmission schedule with the most evenly distributed rates for all the time slots.

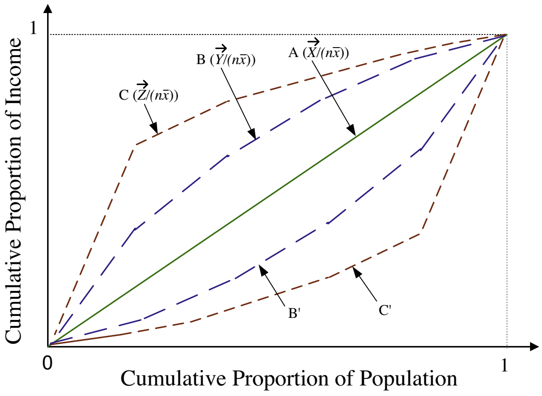


Fig. 3. Illustrating the Lorenz curve.

4. Power allocation algorithms

Based on the stochastic majorization interpretation of Problem (10) and the Schur-convex property of its objective function, we first develop a power optimal algorithm (PMA) for the case of relaxed peak power constraint. We prove its optimality and the uniqueness of the global optimal, as well as the equivalence of power optimal and smoothness optimal. We then describe a heuristic algorithm for the case of multiple videos coupled with the peak power constraint.

4.1. Optimal algorithm for Problem (10)

4.1.1. Power minimization algorithm

From Section 3.1, an evenly distributed rate vector $\vec{C}_n^{opt} = [\Omega_n/(T_n\tau), \cdots, \Omega_n/(T_n\tau)]$ is majorized by all feasible schedules, i.e., $\vec{C}_n^{opt} \prec \vec{C}_n^i$, for all i. However, due to the high variability of VBR video frames, limited playout buffer size, random path gains and noise powers, $\vec{C}_n^{opt}$ may not always be feasible. In general, each feasible schedule is piece-wise linear with a set of rate change points, where the rate is increased or decreased to prevent buffer underflow or overflow. $\vec{C}_n^{opt}$ is a special case with no such rate change points.

The algorithm in Table 2, termed PMA, can generate a piece-wise linear schedule, while keeping each piece as long as possible and rate variation as small as possible. The operation of the algorithm is illustrated in Fig. 4. Starting from t_{start} (e.g., h_1 in Fig. 4), PMA first computes two probe lines:

- One through the starting point and one of the future corner points of $B_n(t)$, which can go the furthest into the future without causing buffer underflow or overflow (e.g., lines h_1h_2 in Case 1

Table 2
Power minimization algorithm (PMA-1).

```
The BS obtains $b_n, D_n, B_n$, and $B_w$ for every session $n$;
$t = 1, t_{start} = 0, t_{stop} = t_{c_1} = t_{c_2} = 1, C_{min} = 0,$
$C_{max} = \infty$;
DO:
  Compute $C_{max}(t)$ and $C_{min}(t)$ over interval $[t_{start}, t]$;
  IF $(C_{min} \leqslant C_{min}(t) \& C_{min}(t) \leqslant \min\{C_{max}, C_{max}(t)\})$
    $C_{min} = C_{min}(t)$ and $t_{c_1} = t$;
  END IF
  IF $(C_{max} \geqslant C_{max}(t) \& C_{max}(t) \geqslant \max\{C_{min}, C_{min}(t)\})$
    $C_{max} = C_{max}(t)$ and $t_{c_2} = t$;
  END IF
  IF $(C_{min} > \min\{C_{max}, C_{max}(t)\})$
    Select $C_{min}$ from $t_{start}$ to $t_{stop} = t_{c_1}$;
  ELSEIF $(C_{max} < \max\{C_{min}, C_{min}(t)\})$
    Select $C_{max}$ from $t_{start}$ to $t_{stop} = t_{c_2}$;
  ELSE
    $t = t + 1$;
    CONTINUE;
  END IF
  $t_{start} = t_{stop}, t_{stop} = t_{c_1} = t_{c_2} = t_{start} + 1,$
  $t = t_{start} + 1, C_{min} = 0, C_{max} = \infty$;
WHILE (some time slots are not assigned with a rate)
DO:
  Measure channel gain of the time slot;
  Compute power using (6);
  Transmit the video data using the computed powers;
WHILE (more video frames to transmit)
```

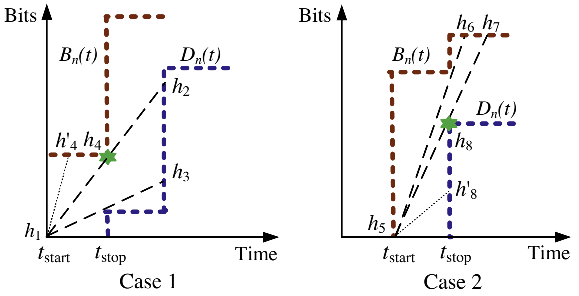


Fig. 4. Determine the next transmission rate.

and h_5h_6 in Case 2 of Fig. 4). The rate of this probe line is $C_{max}(t) = \frac{B_n(t)-X_n(t_{start})}{t-t_{start}}$.

- The other through the starting point and one of the future corner points of $D_n(t)$, which can go the furthest into the future without causing buffer underflow or overflow (e.g., lines h_1h_3 in Case 1 and h_5h_7 in Case 2 of Fig. 4). The rate of this probe line is $C_{min}(t) = \frac{D_n(t)-X_n(t_{start})}{t-t_{start}}$.

All feasible transmission curves should lie in between these two probe lines in order to go that far. Furthermore, when the probe lines end, they hit *both* on either $B_n(t)$ or $D_n(t)$. Otherwise, we can always adjust one of the probe lines to make them go even further into the future. For example, see lines h_1h_3 and $h_1h'_4$ in Case 1 of Fig. 4. We can use line h_1h_2, which goes further into the future, to replace line $h_1h'_4$, and both probe lines hit $D_n(t)$ eventually (also see lines h_5h_6 and $h_5h'_8$ in Case 2).

If both probe lines hit $D_n(t)$ (i.e., Case 1 in Fig. 4), any feasible schedule for this interval will also hit $D_n(t)$, since they must lie in between the two probe lines. We then trace back the upper probe line (i.e., line h_1h_2) to find the latest time when the buffer is full (i.e., point h_4 at time t_{stop}). Then segment h_1h_4 will be chosen as the transmission schedule for this interval, with rate $\frac{B_n(t_{stop})-X_n(t_{start})}{t_{stop}-t_{start}}$.

If both probe lines hit $B_n(t)$ (i.e., Case 2 in Fig. 4), any feasible schedule for this interval will also hit $B_n(t)$. We then trace back the lower probe line (i.e., line h_5h_7) to find the latest time when the buffer is empty (i.e., point h_8 at time t_{stop}). Then segment h_5h_8 will be chosen as the transmission schedule for this interval, with rate $\frac{D_n(t_{stop})-X_n(t_{start})}{t_{stop}-t_{start}}$.

After the transmission schedule for $[t_{start}, t_{stop})$ is determined, we set $t_{start} = t_{stop}$ and repeat the above procedure to find the schedule for the next time interval. In Table 2, the algorithm probes for the longest feasible rate starting from t_{start} in Steps 4–10. In Steps 11–14, the transmission rate for the interval $[t_{start}, t_{stop})$ is determined depending on which of the two cases it is as illustrated in Fig. 4. Steps 16–17 are for the case that the rate does not change in the time slot. Step 19 resets the variables to start the computation for the next segment of $X_n(t)$. Finally, Steps 21–23 transmit the frames following the computed schedule.

4.1.2. Optimality proof

We next show that the algorithm given in Table 2 computes the optimal solution to Problem (10).

Theorem 2. *The power minimization algorithm PMA is optimal to Problem (10).*

Proof. The proof is similar to that in [9] for smoothing a VBR video over a constant capacity channel. The schedule computed by PMA, $\vec{C}^*_n$, could be a straight line or in the general case, consist of one or more convex and concave segments. If $\vec{C}^*_n$ is a straight line, it is obvious that $\vec{C}^*_n \prec \vec{C}^i_n$ for any other $\vec{C}^i_n$ (see Fig. 3) and it is power optimal. In the general case, we need to show $\vec{C}^*_n \prec \vec{C}^i_n$, for all i in every convex or concave segment. Then according to Lemma 1, we have $\vec{C}^*_n \prec \vec{C}^i_n$ for all i and it is optimal.

Let $\vec{C}_n$ denote an arbitrary feasible schedule. We introduce an auxiliary schedule $\vec{C}^1_n$, which intersects with $\vec{C}^*_n$ at all its rate change points in every convex segment, and with $\vec{C}_n$ at all its rate change points in every concave segment, as shown in Fig. 5.

First, we prove that $\vec{C}^*_n \prec \vec{C}^1_n$. For a convex segment of $\vec{C}^*_n$, because $\vec{C}^1_n$ intersects with $\vec{C}^*_n$ at all the rate change points of $\vec{C}^*_n$, we have $\vec{C}^*_n = \vec{C}^1_n$ in all the convex segments. For a concave segment of $\vec{C}^*_n$, the endpoints of the concave segment should be the last (first) rate change point of the previous (next) convex segment, where $\vec{C}^*_n$ intersects with $\vec{C}^1_n$. The rate change points within the concave segment are all on $D_n(t)$, as in PMA. Therefore, $\vec{C}^1_n$ is an outer concave curve above $\vec{C}^*_n$ (or, it is farther away from the straight line A in Fig. 3) in this segment. From the discussion of Fig. 3, we have $\vec{C}^*_n \prec \vec{C}^1_n$ for all the concave segments. It follows that $\vec{C}^*_n \prec \vec{C}^1_n$ according to Lemma 1.

We next prove that $\vec{C}^1_n \prec \vec{C}_n$. For a convex segment of $\vec{C}^*_n$, the endpoints of the convex segment should be the last (first) rate change point of the previous (next) concave segment, where $\vec{C}_n$ intersects with $\vec{C}^1_n$. The rate change points of $\vec{C}^1_n$ (or, of $\vec{C}^*_n$) in the convex segment are all on $B_n(t)$. Therefore, $\vec{C}_n$ is an outer convex curve below $\vec{C}^1_n$ in this segment. From the discussion of Fig. 3, it follows that $\vec{C}^1_n \prec \vec{C}_n$ in all the convex segments. In a concave segment, we have either $\vec{C}_n = \vec{C}^1_n$ or $\vec{C}_n \prec \vec{C}^1_n$, because $\vec{C}^1_n$ intersects with $\vec{C}_n$ at each rate change point. Thus, we obtain $\vec{C}^1_n \prec \vec{C}_n$ for all the concave segments, and $\vec{C}^1_n \prec \vec{C}_n$ according to Lemma 1.

Finally we have $\vec{C}^*_n \prec \vec{C}^1_n \prec \vec{C}_n$. Theorem 1 states that Problem (10) is Schur-convex and order preserving. It follows from Fact 1 that $\vec{C}^*_n$ is optimal to Problem (10). □

Corollary 2.1. *The power optimal transmission scheme $\vec{C}^*_n$ is unique for given $B_n(t)$ and $D_n(t)$.*

Proof. Suppose $\vec{C}^*_n$ is not unique. Then there exists $\vec{C}'_n \prec \vec{C}^i_n$, for all i, and $\vec{C}'_n \neq \vec{C}^*_n$. $\vec{C}'_n$ must have a different set of rate change points from that of $\vec{C}^*_n$. According to the proof of Theorem 2, we can construct an auxiliary schedule $\vec{C}^1_n$, such that $\vec{C}^*_n \prec \vec{C}^1_n \prec \vec{C}'_n$, which contradicts the assumption that $\vec{C}'_n$ is optimal. □

Corollary 2.2. *The computational complexity of Algorithm PMA is $\mathcal{O}(T_n^2)$.*

Proof. In the worst case, the PMA algorithm computes the optimal schedule for each time slot by probing the full length of the remaining video sequence (as in Steps 4–12 in Table 2). The worst case execution time is $\sum_{i=T_n}^{1} i = \frac{T_n(T_n+1)}{2} \Rightarrow \mathcal{O}(T_n^2)$. □

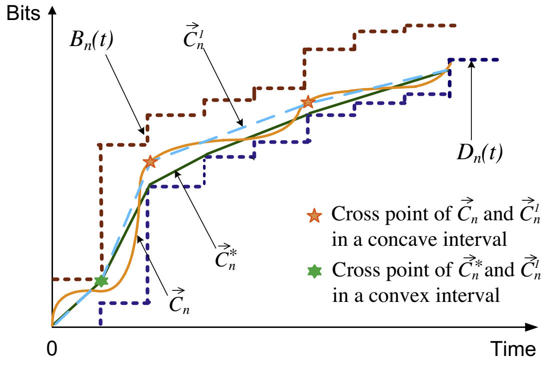

Bits
$B_n(t)$
$\vec{C}^1_n$
$D_n(t)$
$\vec{C}^*_n$
$\vec{C}_n$
Cross point of $\vec{C}_n$ and $\vec{C}^1_n$ in a concave interval
Cross point of $\vec{C}^*_n$ and $\vec{C}^1_n$ in a convex interval
0
Time

Fig. 5. Relationship among $\vec{C}^*, \vec{C}_1$ and $\vec{C}$.

Note that Algorithm PMA is executed during the session setup time. It only incurs a small initialization delay. In our simulations with VBR video traces, we find the execution time is usually negligible. When the channel statistics are changed (i.e., due to handoff), the schedule will be recomputed for the remaining video frames.

Corollary 2.3. *The power optimal transmission schedule $\vec{C}_n^*$ is also the smoothest one among all feasible schedules.*

Proof. To evaluate the smoothness of a transmission schedule $\vec{C}$, the following smoothness utility function can be used:

$$U(\vec{C}) = \sum_{t=1}^{T_n}([c(t) - \bar{c}]/T_n), \qquad (14)$$

where $\bar{c} = \sum_{t=1}^{T_n} c(t)/T_n$ is the average rate. This is a continuous symmetric convex function $U : \mathcal{R}^{T_n} \rightarrow \mathcal{R}$. From Fact 1, U is Schur-convex and order preserving. The optimal power transmission schedule $\vec{C}_n^*$ satisfies $\vec{C}_n^* \prec \vec{C}_n^i$ for all i. Therefore, it also achieves the minimum value for $U(\cdot)$. □

4.2. Multiuser video transmissions

We now consider Problem (7) to compute transmission schedules for N VBR video sessions, which are coupled by the peak power constraint (9). Due to the peak power constraint and random channel gains, the individually calculated transmit powers may violate (9) in some time slots. The problem is further complicated by the random channel gains, which is not available a priori (except for the statistics of the channels).

To solve Problem (7), we develop a heuristic algorithm, termed PMA-*m*, as presented in Table 3. The PMA-*m* algorithm uses PMA to compute transmission schedules for all active users. Then based on current channel state information, it computes the power needed to achieved the rate for each user, and checks the peak power constraint $\sum_{n \in \mathcal{U}_n} P_n(t) \leqslant \overline{P}$. If the constraint is not violated, each user's video data will be transmitted at the computed power. Otherwise, as in Steps 4–10, PMA-*m* selects those users who will not have buffer underflow if their transmissions are suspended in the following time slot, and sort them in the deceasing order of their required powers. Starting with the first user, PMA-*m* decreases the powers of the users in the list; if the first user's power reaches 0 W but the peak power constraint is till not satisfied, PMA-*m* starts to reduce the power of the second user in the list; and so forth until the peak power constraint is satisfied.

In some extremely severe channel conditions, the total power $\overline{P}$ cannot even support the minimum required bit rate for all the users. Some users have to be paused and the current frames be discarded. The corresponding playout of such a user will be frozen until the next time slot. Finally, the transmission schedules for the suspended users will be recomputed using PMA as in Step 11 and the above procedure is repeated.

Table 3
Power minimization algorithm for multiuser videos (PMA-*m*).

```
1   Run PMA to compute transmission schedules for all sessions;
2   DO:
3     Measure channel gains of the current time slot and compute
      the transmit powers using (6);
4     IF (constraint (9) is violated)
5       Select the users who won't have underflow if they are not
        scheduled to transmit in this time slot;
6       Sort these selected users in decreasing order of powers;
7       DO:
8         Decrease the power of the selected users by the order;
9       WHILE (constraint (9) is not satisfied)
10    END IF
11    Transmit the videos following the schedules;
12    Recalculate transmission scheme for the paused sessions for
      the next time interval;
13  WHILE (more video frames to transmit)
```

4.3. Application to interactive video streaming

The ubiquitous spread of mobile devices and trend of multimedia applications require the interactive service be supported [4–6]. Thus, it is necessary to investigate how to apply the proposed schemes to the case of interactive video streaming. Interactive video is a relatively new and still evolving technology with a broad scope. We focus on the three interactive video streaming related typical scenarios in the following, and show that the proposed schemes are applicable for these scenarios for improved performance.

First, for quick response to user inputs, many interactive video streaming applications have *tight delay requirements*. Such stringent delay requirements have two implications: (i) unlike stored video, not all the future frame sizes are known now; only the frames sizes for a short *look-ahead period* (LAP) are known. (ii) the playout buffer sizes cannot be large, since a large buffer usually introduces large delay. Clearly, the proposed schemes can be applied to the look-ahead time period for which the future frame sizes are known, to compute a schedule for the near future. Furthermore, given the small playout buffer size, usually a chosen transmission rate won't last very long into the future before it hits either the cumulative overflow curve or the cumulative consumption curve (see Fig.4. Therefore the impact of limited look-ahead period would be small or moderate at best.

Second, many interactive video applications support *VCR controls* [7]. For example, a user may slide the progress bar of the video player to replay or skip a part of the video. This case is equivalent to a change in the cumulative overflow and consumption curves. The proposed algorithms will seek to the new start frame that the user requires and recalculate the transmission schedules for the following frames.

Third, in both "*exploratory*" online interactive videos (where a user can move through different locations in a space or view an object from different angles) and *video click throughs* [8] (where a user can click objects in the video that are linked to other contents), new data will be transmitted after each user input. These are equivalent to the case of VCR controls. A new set of cumulative overflow and consumption curves will be delivered (derived from the new data requested) and new schedules computed.

In Section 5.3, we evaluate the performance of the proposed schemes under the above interactive video streaming scenarios. Our simulation results show that the proposed schemes still achieve considerable power savings and better video quality than a conventional scheme for interactive videos.

5. Simulation study

5.1. Simulation settings

We study the performance of the proposed optimal power control algorithms with trace-driven simulations. We simulate the downlink of a cell with 1 mile radius. The channels are assumed to be orthogonal, each with B_w =1 MHz bandwidth. We assume that bit errors can be effectively corrected by error correction codes. The path gain averages are $\overline{G}_n = d_n^{-4}$, where d_n is the physical distance from the BS to user n. We assume log-normal fading with zero mean and 8 dB standard deviation. The device temperature is 290 K and the equivalent noise bandwidth is B_w =1 MHz.

In the simulations, the BS streams several movies to active users in the cell. We use six VBR video traces in the simulations: *Star*

Wars, *NBC News*, *Tokyo Olympics*, *Terminator 2*, *From Mars to China*, and *Sony Demo*. The video traces are obtained from the Video Trace Library from Arizona State University, Tempe, AZ, USA [14]. The statistics of the six video traces are summarized in Table 4.

5.2. Simulation results for stored video streaming

We first investigate the performance of the power optimal algorithm for stored VBR videos. In the simulation, the BS streams 3000 frames of a video sequence to each mobile user located at different distances to the BS. The cumulative consumption, overflow and transmission curves of the *Star Wars* video session are plotted in Fig. 6. It can be seen that the transmission schedule always lie in between the cumulative consumption and overflow curves, indicating that there is no playout buffer underflow or overflow events in this simulation.

We next compare the optimal power algorithm with a conventional transmission scheme with respect to the average power consumption at the BS. The "*lazy*" scheme presented in [15] is simulated for comparison purpose. This scheme is an ON–OFF scheme and transmits the video frames as late as possible before its playout deadline with the maximum link speed, which minimizes the required client buffer size. In wireless VBR streaming, the maximum link speed varies from time to time due to channel interference and channel fading, so the original lazy scheme cannot be applied directly. Thus, we consider the wireless modification of the original lazy algorithm: in each time slot, the conventional scheme only transmits the video data that is needed by the decoder at the end of the time slot. It is called the "*lazy*" scheme due to this minimal transmission strategy. In particular, the lazy scheme achieves a cumulative transmission curve that connects all the corner points of $D_n(t)$. Intuitively, such lazy approach should be energy efficient since it always transmits the minimal amount of data as needed. However, we will see that the proposed algorithms outperform this lazy approach in the simulations.

Table 4
VBR video trace statistics [14].

Video trace	Average rate	Frame rate	Average PSNR
Star Wars	331,681 b/s	30 f/s	44.62 dB
NBC News	784,840 b/s	30 f/s	38.80 dB
Tokyo Olympics	509,191 b/s	30 f/s	41.46 dB
Terminator 2	5,085,453 b/s	30 f/s	43.92 dB
From Mars to China	4,849,711 b/s	30 f/s	39.26 dB
Sony Demo	5,803,650 b/s	30 f/s	44.07 dB

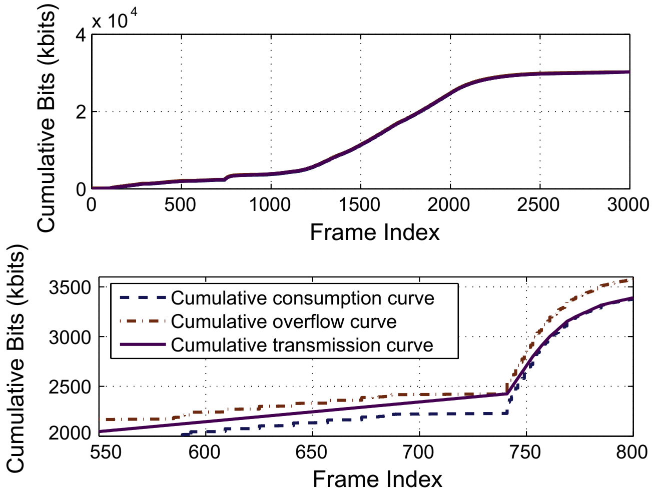


Fig. 6. Transmission curves of *Star Wars* computed by PMA-1 for stored VBR video.

In Fig. 7, we plot the average power consumption achieved by the two schemes for increased distance to the BS. Each point in the figure is the average of 10,000 simulation runs. The 95% confidence intervals are plotted as vertical bars in the figure, which are all very small. It can be seen that the proposed algorithm PMA-1 outperforms the conventional scheme for the entire range of distances examined. When the distance to BS is small, both schemes use small transmit powers and the power savings are not very big. However, when the distance is increased, channel fading has a bigger impact on interference and channel capacity. The proposed algorithm achieves considerable power savings than the conventional scheme. When the distance is 1,600 m, the total power of the proposed scheme is only 46.62% of that of the conventional approach, corresponding to a 54.34% normalized improvement.

We further investigate in more detail the difference of the transmissions between the two schemes. The position of a mobile node is set to 1000 m from the BS. The first 3000 frames of the *Star Wars* movie are transmitted to the node using the PMA-1 scheme and conventional scheme, respectively. Fig. 8 shows the cumulative power consumption for the first 1000 frames, while the energy consumption for each video frame is plotted in Fig. 9 for frames in [200,250]. We observe that at the beginning, the "*lazy*" scheme archives smaller power consumption than the PMA scheme, due to the fact that it only transmits the the minimum amount of required

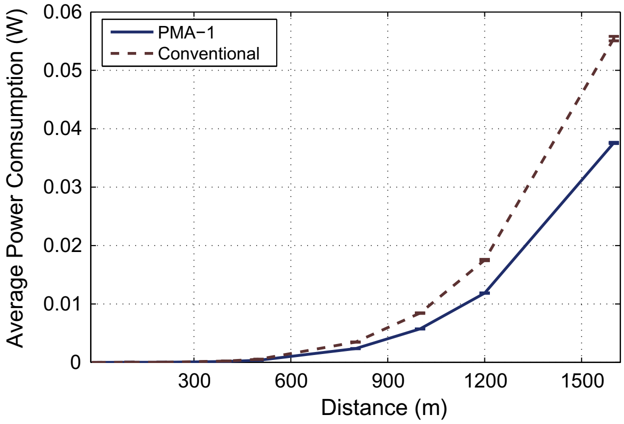


Fig. 7. Average power consumption achieved by PMA-1 and the conventional scheme.

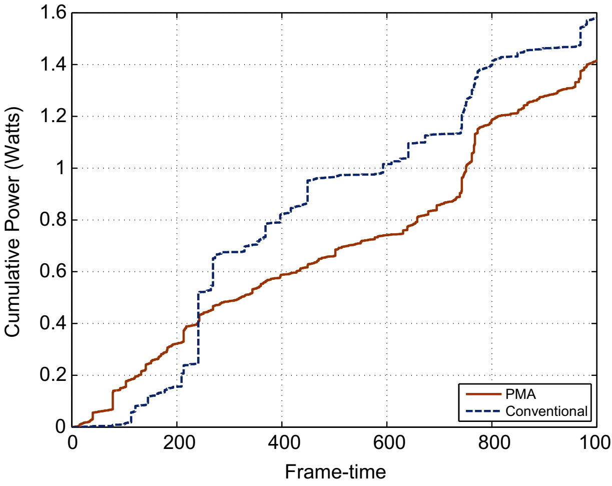


Fig. 8. Cumulative power consumptions achieved by PMA-1 and the conventional scheme.

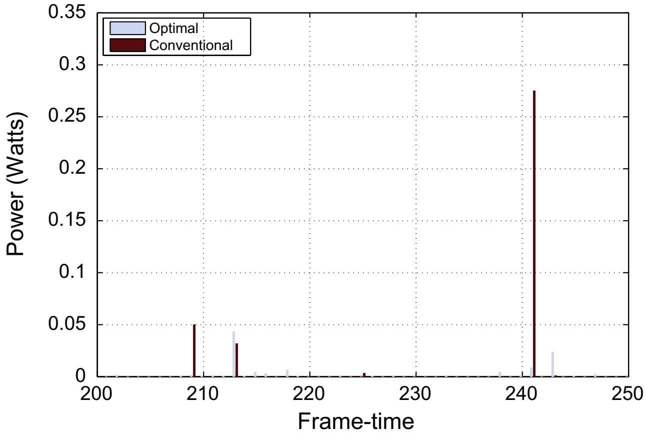
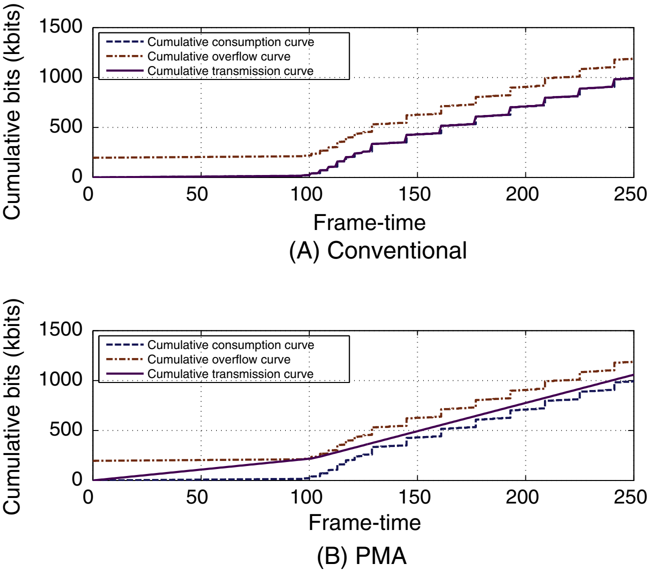


Fig. 9. Power consumption for each frame achieved by PMA-1 and the conventional scheme.



(A) Conventional

(B) PMA

Fig. 10. Transmission curves achieved by PMA-1 and the conventional scheme.

frames in each time slot. However, the transmission of frame 241 of the conventional scheme generates a sharp power increase, because it encounters a large frame as well as bad channel condition, as indicated in Fig. 9. The transmission curves of the two schemes are plotted in Fig. 10 for the first 250 frames for the two schemes. Although the conventional scheme archives smaller power consumption frame by frame for about 80% of the 3000 frames, the average power consumption of the proposed scheme during the entire period (0.0055 W) is much smaller than that of the conventioanal scheme (0.0141 W). In summary, the "*lazy*" scheme only uses the current video and channel status, and transmits only the minimum amount of required video data, It does not effectively utilize the playout buffer capacity. Thus during the entire transmission period, the cumulative power may increased due to some large frames and bad channel conditions. On the contrary, the proposed scheme aims at minimizing the total average transmission power during the entire period. Thus, it achieves considerable power savings comparing to the conventional scheme.

We also obtain the average execution time of the proposed algorithm, under the same setting but for 20,000 *Star Wars* frames. We find that the average execution time is about 0.06 s on an IBM Laptop with Intel T2400 1.83 GHz processor and 2 GB RAM.

Finally, we examine the buffer underflow events, where PMA-*m* is used for multiuser VBR videos that are coupled with the peak power constraint. The following scenarios are simulated:

- *Scenario I*: $\overline{P} = 1$ W; the movies are *Star Wars*, *NBC News*, and *Tokyo Olympics*; 50 mobile users; $B_w = 1$ MHz;
- *Scenario II*: Same setting as Scenario I, except that $B_w = 125$ KHz;
- *Scenario III*: $\overline{P} = 10$ W; the HD movies are *Terminator 2*, *From Mars to China*, and *Sony Demo*; 20 mobile users; $B_w = 1$ MHz.

The HD movies have larger frame sizes and higher variability in frame sizes. The buffer underflow rates are presented in Table 5, each being the ratio of the number of underflow frames over the total number of frames. PMA-*m* achieves considerably lower underflow rates for all the three scenarios. In particular, the PMA-*m* underflow rates are 0.056%, 13.60%, and 14.26% of those of the conventional scheme, indicating superior viewer experience. Therefore, PMA-*m* achieves not only considerable energy savings, but also much better video quality at the mobile users.

5.3. Simulation results for interactive video streaming

In this section, we study the performance of the proposed algorithms for interactive video streaming. First, we simulate the interactive real-time video streaming with stringent delay requirements and small playout buffer sizes. The same simulation settings are used. All the positions of mobile nodes are randomly generated in the cell. We apply the proposed PMA-*m*, but only assume only the frame sizes in a small LAP are known.

The transmission curves of VBR movie *NBC News* are plotted in Fig. 11 for the first 100 frames, where the length of the LAP is 16 frames. We may observe that the proposed algorithm PMA-*m* is executed piecewisely for each block of LAP frames, where LAP = 16. For this range of frames, there is neither playout buffer overflow nor playout buffer underflow occurs. We then compute the number of the underflow frames over the total number of frames, as presented in Table 6. The PMA-*m* still archive the

Table 5
Playout buffer underflow rates for PMA-*m* and the conventional scheme for stored VBR videos.

	Scenario I (%)	Scenario II (%)	Scenario III (%)
PMA-*m*	0.0005	1.8	1.66
Conventional	0.89	13.24	11.64

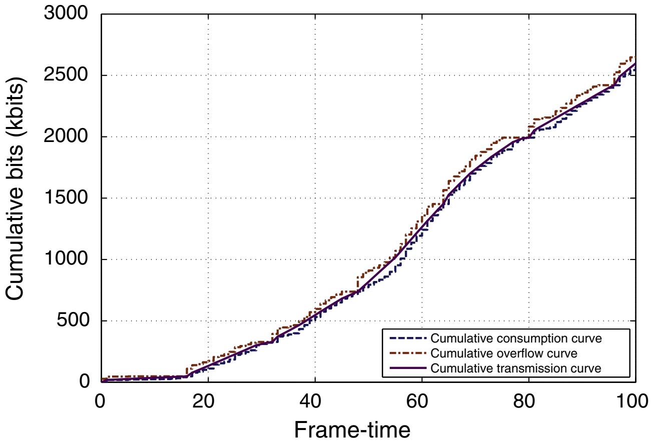


Fig. 11. Transmission curves for real-time interactive video streaming: *NBC News* and LAP = 16.

Table 6
Playout buffer underflow rates for interactive VBR videos with different LAPs.

	Scenario I (%)	Scenario II (%)	Scenario III (%)
	LAP = 16		
PMA-*m*	0.387	5.969	4.766
Conventional	0.711	12.446	12.495
	LAP = 8		
PMA-*m*	0.377	5.690	4.894
Conventional	0.620	11.409	9.645

underflow rate that are 54.56%, 47.96%, 38.15% of those of the conventional scheme in this case.

To illustrate the impact of the length of LAP, we further decrease it to 8 frames and then run the simulations with random deployed mobile nodes. The playout buffer underflow rates are presented in Table 6. For the halved delay requirement, naturally the proposed scheme's performance is slightly degraded due to limited information about the video frame sizes. However, the PMA-*m* scheme still archives underflow rates that are 60.90%, 49.88%, 50.74% of those of the conventional scheme, indicating considerably superior viewer performance over the conventional "lazy" approach.

Finally, we demonstrate the application of the proposed schemes to VCR control in interactive video streaming. We assume that after 10 s of streaming the VBR video (i.e, corresponding to 300 frames), the user skips the next 10 s of the video, and then resumes the video playout from 20 s. We plot the dynamics in the transmission/schedule curves of the *Star Wars* video in Fig. 12. Comparing the curves with the original non-skipped transmissions in Fig. 13, we observe that after playing out the 300th frame, the frame from 301 to 600 are skipped by user's operation. Then the frame 601 is moved to the time-slot 301 and a new transmission schedule is computed for the following frames.

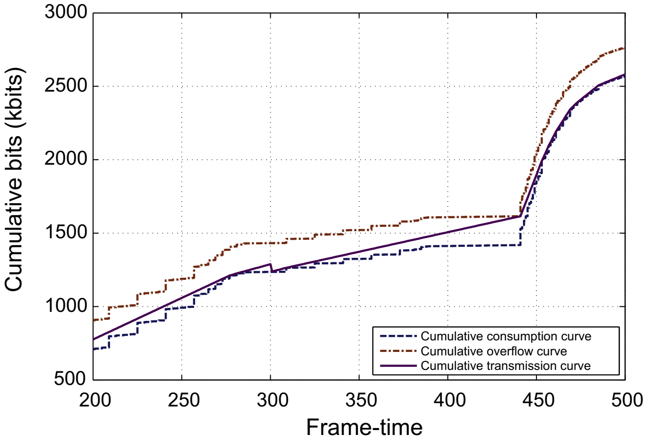


Fig. 12. Transmission curves after the user skips the frames in [10 s, 20 s]: *Star Wars*.

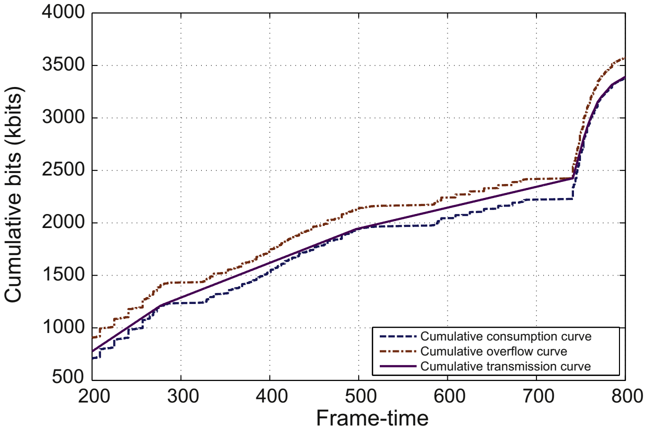


Fig. 13. Transmission curves for the original non-skipped video streaming: *Star Wars*.

6. Related work

VBR video over wired networks is well-studied under two types of models: *statistical* or *deterministic* models. The former models VBR traffic with stochastic process and observed long-range-dependence in VBR trace [2,16], which provides valuable knowledge for the nature of VBR traffic. The latter method uses piecewise-constant-rate transmission and transport (PCRTT) method to optimize one or several objects while preserving the continuous video playout [17,18,9]. In an interesting work [9], Salehi, et al. applied majorization to VBR video smoothing and developed a smoothness optimal algorithm. The proof of Theorem 2 follows a similar approach as in [9]. These prior work are based on the assumptions of a single video session and constant rate channels, which may not be directly applied to the case of wireless networks.

There are two closely related pieces of work on VBR video over wireless channels [19,20]. Both papers are focused on one VBR stream over a given time-varying wireless channel. In [19], it was shown that the separation between a delay jitter buffer and a decoder buffer is in general suboptimal, and several critical parameters are derived for the system. In [20], the authors studied the frequency of jitters under both network and video system constraint and provided a framework for quantifying the trade-offs among several system parameters.

In this paper, we consider multiuser VBR video streaming within a cellular network with orthogonal channels. We jointly consider power control, video traffic, and video palyout information for power minimization. Our stochastic majorization theory based approach is quite different from the prior works [19,20], which allow us to develop effective algorithms with low complexity and proven optimality.

7. Conclusion

In this paper, we studied the problem of downlink multiuser VBR video streaming in cellular networks. Our formulation takes into account the interactions among power control, fading channels, VBR video traffic and playout characteristics. We formulate a constrained stochastic optimization problem aiming to minimize the BS power consumption and to avoid playout buffer overflow or underflow. We developed majorization-based algorithms to solve the formulated problem. For the case of large peak power constraints, we prove the optimality of the proposed algorithm and the uniqueness of the global optimal, as well as the equivalence of power optimal and smoothness optimal. For the case of multiple videos coupled with the peak power constraint, we develop an effective heuristic algorithm that selectively suspends some video sessions when the peak power constraint is violated. The superior performance of the proposed algorithms over a conventional scheme is validated with trace-driven simulations.

Acknowledgment

This work is supported in part by the US National Science Foundation (NSF) under Grants CNS-0953513, ECCS-0802113, CNS-1145446 and IIP-1127952, and DUE-1044021, and through the

NSF Wireless Internet Center for Advanced Technology at Auburn University (Grant IIP-0738088). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF.

References

[1] Y. Huang, S. Mao, Downlink power control for variable bit rate video over multicell wireless networks, in: Proceedings IEEE INFOCOM'11, Shanghai, China, April. 2011, pp. 2561–2569.

[2] M.W. Garrett, W. Willinger, Analysis modeling and generation of self-similar VBR video traffic, ACM SIGCOMM Comput. Commun. Rev. 24 (4) (1994) 269–280.

[3] A.W. Marshall, I. Olkin, Inequalities: Theory of Majorization and Its Applications, Academic Press, New York, NY, 1979.

[4] A. Naman, D. Taubman, JPEG2000-based scalable interactive video (JSIV), IEEE Trans. Image Process. 20 (5) (2011) 1435–1449.

[5] W.-K. Liao, Y.-H. Lai, Type-aware error control for robust interactive video services over time-varying wireless channels, IEEE Trans. Mobile Comput. 10 (1) (2011) 136–145.

[6] G. Cheung, A. Ortega, N.-M. Cheung, Interactive streaming of stored multiview video using redundant frame structures, IEEE Trans. Image Process. 20 (3) (2011) 744–761.

[7] A. Dan, P. Shahabuddin, D. Sitaram, D. Towsley, Channel allocation under batching and vcr control in movie-on-demand servers, J. Parallel Distrib. Comput. 30 (1995) 168–179.

[8] buto.tv, Feature: Clickable videointeractive, clickable video that anyone can build in minutes, Available: <get.buto.tv/features/clickable-video/>.

[9] J. Salehi, Z.-L. Zhang, J. Kurose, D. Towsley, Supporting stored video reducing rate variability and end-to-end resource requirements through optimal smoothing, IEEE/ACM Trans. Netw. 6 (4) (1998) 397–410.

[10] J. Lee, R. Mazumdar, N. Shroff, Downlink power allocation for multi-class wireless systems, IEEE/ACM Trans. Netw. (13 4) (2005) 854–867.

[11] S. Ulukus, A. Yener, Iterative transmitter and receiver optimization for CDMA networks, IEEE Trans. Wireless Commun. 3 (6) (2004) 1879–1884.

[12] E. Jorswieck, H. Boche, Optimal transmission strategies and impact of correlation in multiantenna systems with different types of channel state information, IEEE Trans. Signal Process. 52 (12) (2004) 3440–3453.

[13] B.C. Arnold, Majorization and the Lorenz Order: A Brief Introduction, Springer-Verlag, New York, NY, 1987.

[14] M. Reisslein, Video trace library, Arizona State University, Available <trace.eas.asu.edu/>.

[15] S. Sen, D. Towsley, Z. Zhang, J.K. Dey, Optimal multicast smoothing of streaming video over the internet, IEEE J. Sel. Areas Commun. 20 (7) (2002) 1345–1359.

[16] J. Beran, R. Sherman, M. Taqqu, W. Willinger, Long-range dependence in variable-bit-rate video traffic, IEEE Trans. Commun. 43 (2-4) (1995) 1566–1579.

[17] S. Liew, H. Chan, Lossless aggregation a scheme for transmitting multiple stored VBR video streams over a shared communications channel without loss of image quality, IEEE J. Sel. Areas Commun. 15 (6) (1997) 1181–1189.

[18] W.-C. Feng, M. Liu, Critical bandwidth allocation techniques for stored video delivery across best-effort networks, in: Proceedings IEEE ICDCS'00, Taipei, Taiwan, April 2000, pp. 56–63.

[19] T. Stockhammer, H. Jenkac, G. Kuhn, Streaming video over variable bit-rate wireless channels, IEEE Trans. Multimedia 6 (2) (2004) 268–277.

[20] G. Liang, B. Liang, Balancing interruption frequency and buffering penalties in VBR video streaming, in: Proceedings IEEE INFOCOM'07, Anchorage, AK, May 2007, pp. 1406–1414.