Scalable Video Multicast in Cognitive Radio Networks

Donglin Hu, Shiwen Mao, *Senior Member, IEEE*, Y. Thomas Hou, *Senior Member, IEEE*, and Jeffrey H. Reed, *Fellow, IEEE*

***Abstract*—We investigate the problem of scalable video multicast in emerging cognitive radio (CR) networks. Although considerable advances have been made in CR research, such important problems have not been well studied. Naturally, "bandwidth-hungry" multimedia applications are excellent candidates for fully capitalizing the potential of CRs. We propose a cross-layer optimization approach to multicast video in CR networks. Specifically, we consider an infrastructure-based CR network collocated with N primary networks and model CR video multicast over the N channels as a mixed integer nonlinear programming (MINLP) problem. The objective is three-fold: to optimize the overall received video quality; to achieve proportional fairness among multicast users; and to keep the interference to primary users below a prescribed threshold. We propose a *sequential fixing algorithm* and a *greedy algorithm* to solve the MINLP, while the latter has low complexity and proven optimality gap. Our simulations with MPEG-4 fine grained scalability (FGS) video demonstrate the efficacy and superior performance of the proposed algorithms.**

***Index Terms*—Video multicast, cognitive radio networks, dynamic spectrum access, cross-layer optimization**

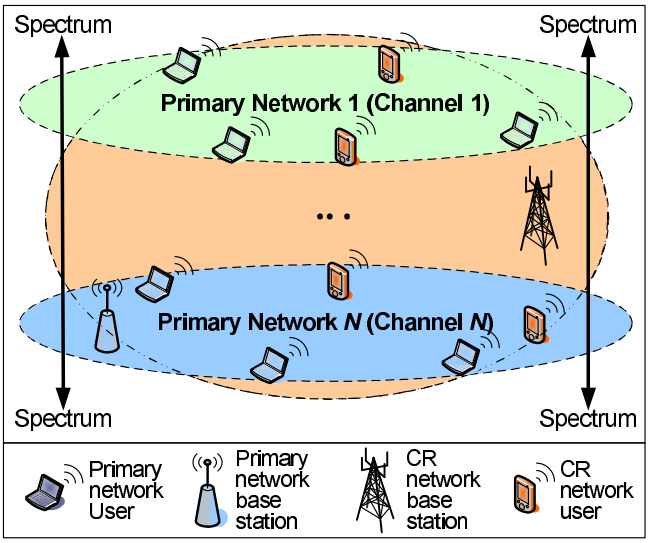


Fig. 1. Illustration of the network architecture considered in this paper: an infrastructure-based CR network collocated with N primary networks.

I. Introduction

A *COGNITIVE RADIO* (CR) is a frequency-agile wireless communication device that enables *dynamic spectrum access* (DSA). CR represents a significant paradigm change in spectrum regulation and utilization. Its high potential has stimulated a flurry of activities in engineering, economics, and regulatory communities in search of better spectrum management and sharing policies [1]–[3]. Since the focus of CR research has been mainly on DSA, the application layer performance over CR networks has not yet been well studied. Some important questions, such as *what applications can make efficient use of spectrum whitespace* and *whether existing wireless protocols can provide satisfactory performance*, remain to be answered. Naturally, "bandwidth-hungry" multimedia applications are excellent candidates for fully capitalizing the potential of CRs. It is therefore important to study how the enhanced spectrum usage can benefit multimedia in CR networks.

In this paper, we present a study of optimized real-time video multicast in CR networks. We consider an infrastructure-based CR network collocated with N primary networks, as illustrated in Fig. 1. Each primary network is allocated with a channel. The availability of each channel evolves over time due to primary user transmissions. We consider multicast application due to its generality and bandwidth efficiency. The CR base station exploits spectrum opportunities in the N channels to multicast videos to G multicast groups. In order to accommodate heterogeneous user channels, we consider fine grained scalability (FGS) to encode each video into a base layer and an enhancement layer [4]. With FGS, the enhancement layer can be truncated at any bit location, while the remaining bits are all useful for decoding. Therefore a user can receive a video quality commensurate to its channel condition.

We present a formulation for the CR video multicast problem, taking into account of various cross-layer design factors such as scalable video coding, spectrum sensing, dynamic spectrum access, modulation, scheduling, error control, and primary user protection. The objective is three-fold: (i) to optimize the overall received video quality, (ii) to achieve

Manuscript received 1 March 2009; revised 20 October 2009. This work was supported in part by the US National Science Foundation (NSF) under Grants ECCS-0802113 and CNS-0855251, and through the Wireless Internet Center for Advanced Technology (WICAT) at Auburn University. The research of Y.T. Hou was supported in part by the NSF under Grant CNS-0721570. J.H. Reed's work was supported in part by the Wireless@Virginia Tech Partners Program. This work was presented in part at IEEE INFOCOM'09, Rio de Janeiro, Brazil, Apr. 2009.

D. Hu and S. Mao are with the Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849-5201 (e-mail: dzh0003@auburn.edu, smao@ieee.org).

Y.T. Hou and J.H. Reed are with the Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061 (e-mail: thou@vt.edu, reedjh@vt.edu).

Digital Object Identifier 10.1109/JSAC.2010.100414.

proportional fairness among multicast users, and (iii) to protect primary users from harmful collisions. Unlike prior work on wireless video [5]–[8], the challenge stems from dynamic channel availability processes, tightly coupled design choices, and the need to predict future channel status under the presence of sensing errors for partitioning, modulation and scheduling of real-time video data.

We show that the formulated problem is a mixed integer nonlinear programming (MINLP) problem, which usually has high complexity to solve. However, the partitioning and scheduling of video packets to the multiple channels should be performed in real-time. This is because that the availability of the channels are determined by primary user transmissions, which may occur in any time slot, and that DSA should be unobtrusive to primary users. Any change in channel status will affect both video data partitioning and packet scheduling. The need for real-time execution calls for low-complexity algorithms. For performance, we aim to design algorithms with proven optimality bounds for video applications.

We propose a two-step approach to solve the formulated MINLP problem. For each group of pictures (GoP), we first determine the optimal partition (and thus rates and modulation-coding (MC) schemes) of FGS video data. We present two computationally efficient algorithms for this purpose: (i) a *sequential fixing* (SF) algorithm based on a linear relaxation of the MINLP problem [9], [10]; and (ii) a *greedy algorithm*, termed GRD1, which exploits the inherent priority structure of FGS video and the order of user channels according to their qualities. We show that GRD1 can guarantee a solution that is within a factor of $(1 - e^{-1/2})$ of the global optimal solution, while with a polynomial complexity suitable for execution of each GoP. The computed solution is further adjusted in each time slot based on more recent channel sensing results and feedback, using a *refined greedy algorithm*, termed GRD2, with a further reduced complexity suitable for execution in each time slot. Second, during each time slot, we use a *tile scheduling algorithm*, termed TSA, to assign video packets (the amount of which is determined by GRD1 and GRD2) to available channels, while each channel is accessed with a probability derived from spectrum sensing results. We show that TSA has a low complexity of $O(N \log N)$ and is optimal in terms of maximizing the total utility of the users.

We present simulation results to provide a comparison study with alternative schemes as well as demonstrating the impact of several key design parameters on the overall system performance. We observe excellent performance achieved by the proposed algorithms, with considerable improvement over an alternative equal allocation scheme. We found that the opportunistic spectrum access approach makes FGS video highly robust to sensing errors [2], [11].

The rest of this paper is organized as follows. In Section II, we present the CR video multicast framework. The proposed algorithms are presented in Section III. We show our simulation study in Section IV and discuss related work in Section V. Section VI concludes the paper.

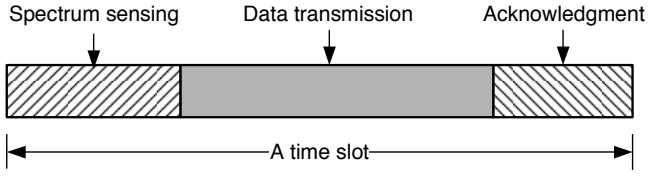


Fig. 2. The structure of a time slot.

II. The CR Video Multicast Framework

A. Network Model

1) Primary Networks: Consider a spectrum band consisting of N channels, each evolving over time independently. For ease of explanation, we assume that the spectrum is continuous. As illustrated in Fig. 1, we assume that the N channels are allocated to N primary networks. For ease of presentation, we assume the primary users access the channels following a synchronous slot structure [2], [12]. The status of each of the N channels evolves following a two-state discrete-time Markov process [2], [13]. The network status in slot t is $\vec{S}(t) = [S_1(t), S_2(t), \cdots, S_N(t)]$, where $S_n(t)$ denotes the status of channel n with idle $(S_n(t) = 0)$ and busy $(S_n(t) = 1)$ states. Let λ_n be the transition probability of remaining in the idle state, and μ_n the transition probability from the busy state to the idle state for channel n.

2) CR Network: We assume a CR network collocated with the N primary networks, within which a base station multicasts G real-time videos to G multicast groups, each having N_g users, $g = 1, 2, \cdots, G$. The base station seeks spectrum opportunities in the N channels. In each time slot t, the base station chooses a set of channels $\mathcal{A}_1(t)$ to sense and a set of channels $\mathcal{A}_2(t)$ to access. The base station has $|\mathcal{A}_1(t)|$ transceivers such that it can sense $|\mathcal{A}_1(t)|$ channels simultaneously. A time slot and channel combination, termed a *tile*, is the minimum unit for resource allocation.

We adopt the same time-slot structure as in [2], [12], which is illustrated in Fig. 2. At the beginning of each time slot, the base station senses channels in $\mathcal{A}_1(t)$ and then chooses a set of available channels for opportunistic transmissions based on sensing results. After a successful transmission, the base station will receive an ACK from the user with the highest SNR in the target multicast group.[1]

Without loss of generality, we assume that each CR network user can access all the N channels. Since it is always desirable to have low hardware requirements, we assume each CR network user has one antenna. We adopt OFDM as multicast technology at the PHY layer. In OFDM, the spectrum is divided into narrow-band channels and the signals are modulated on the channels in the frequency domain. An OFDM frame can be transmitted through one antenna and received through one antenna, since the symbols can be modulated to multiple channels by inverse fast Fourier transform (IFFT). The adaptability of OFDM makes it highly suitable for CR networks [2], [15], [16].

[1]Although ACKs are not adopted in many multicast applications, it has been shown that the *feedback implosion* problem can be effectively solved using properly designed timers [14] (e.g., reversely proportional to channel SINR). The ACKs are important for predicting future channel status (see Section II-B).

B. Spectrum Sensing

Although precise and timely channel state information is desirable for spectrum access and primary user protection, continuous full-spectrum sensing is both energy inefficient and hardware demanding. We assume $|\mathcal{A}_1(t)|$ channels are sensed in each time slot, while sensing is carried out on every $W = N/|\mathcal{A}_1(t)|$ channels [12]. The indices of the channels to be sensed are

$$n = (hW + t) \bmod (N+1), \quad h = 0, 1, \cdots, |\mathcal{A}_1(t)| - 1, \quad (1)$$

where t is the discrete time index. The base station senses the channels in an increasing order. At the beginning of a slot, the base station chooses $|\mathcal{A}_1(t)|$ channels to sense following (1). It then predicts channel status based on sensing results and channel history.

During the sensing process, two kinds of detection errors may occur. When there is a *false alarm*, a spectrum opportunity will be wasted. When there is a *miss detection*, the base station may make a transmission on a busy channel and thus causes collision with primary users. Let ϵ_n and δ_n denote the probabilities of false alarm and miss detection on channel n, respectively. The spectrum sensing performance can be represented by the Receiver Operation Characteristic (ROC) curve, which gives $1 - \delta_n$ as a function of ϵ_n [2], [11]. Let $\overrightarrow{R}(t) = [R_1(t), R_2(t), \cdots, R_N(t)]$ be the sensing outcome in slot t. If channel n is not sensed in time slot t, we have $R_n(t) = -1$. If channel n is sensed in slot t, it has value of either $R_n(t) = 1$ (i.e., sensed busy) or $R_n(t) = 0$ (i.e., sensed idle). The sensing error probabilities for a sensed channel n are $P(R_n(t) = 1|S_n(t) = 0) = \epsilon_n$ and $P(R_n(t) = 0|S_n(t) = 1) = \delta_n$.

Let $\vec{a}(t) = [a_1(t), a_2(t), \cdots, a_N(t)]$ be the *belief vector*, where each element is the conditional probability $a_n(t) = P(S_n(t) = 0|\theta_n(t))$ and $\theta_n(t)$ is defined as the channel n history up to the end of the sensing stage of time slot t. Furthermore, let $\check{a}_n(t) = P(S_n(t) = 0|\check{\theta}_n(t))$ and $\check{\theta}_n(t)$ is the channel n history up to the end of the ACK stage of time slot t. $a_n(t)$ and $\check{a}_n(t)$ are conditional probabilities based on past sensing results and transmission results on channel n, respectively. We have for time slot t

$$\check{a}_n(t) = \begin{cases} 1, & \text{trans. in slot } t \text{ \& ACK received} \\ 0, & \text{trans. in slot } t \text{ \& no ACK received} \\ a_n(t), & \text{no transmission in slot } t. \end{cases} \quad (2)$$

The belief vector $\vec{a}(t)$ can be estimated by considering the following three cases. Case I, channel n is not sensed in time slot t. We need to estimate $a_n(t)$ from the channel n history in the previous time slot. We have

$$\begin{aligned} a_n(t) &= P(S_n(t) = 0|R_n(t) = -1, \theta_n) \\ &= \lambda_n \check{a}_n(t-1) + \mu_n[1 - \check{a}_n(t-1)] = \pi_n(t). \quad (3) \end{aligned}$$

Case II, channel n is sensed in time slot t and the sensing result is $R_n(t) = 0$. The availability of channel n in time slot t is then conditioned on both the channel history and the sensing result. We have

$$\begin{aligned} a_n(t) &= P(S_n(t) = 0|R_n(t) = 0, \theta_n) \\ &= \frac{P(S_n(t) = 0, R_n(t) = 0|\theta_n)}{\sum_{X_n \in \{0,1\}} P(S_n(t) = X_n, R_n(t) = 0|\theta_n)} \\ &= \frac{P(R_n(t) = 0|S_n(t) = 0)P(S_n(t) = 0|\theta_n)}{\sum_{X_n} P(R_n(t) = 0|S_n(t) = X_n)P(S_n(t) = X_n|\theta_n)} \\ &= \frac{\pi_n(t)(1-\epsilon_n)}{\pi_n(t)(1-\epsilon_n) + [1 - \pi_n(t)]\delta_n}. \quad (4) \end{aligned}$$

The third step in (4) is due to the memoryless property of the channel process.

Case III, channel n is sensed in time slot t and the sensing result is $R_n(t) = 1$. Similarly, we can derive the expression for $a_n(t)$ for this case as

$$\begin{aligned} a_n(t) &= P(S_n(t) = 0|R_n(t) = 1, \theta_n) \\ &= \frac{\pi_n(t)\epsilon_n}{\pi_n(t)\epsilon_n + [1 - \pi_n(t)](1 - \delta_n)}. \quad (5) \end{aligned}$$

C. Opportunistic Spectrum Access

Based on spectrum sensing results, the base station determines which channels to access for transmission of video data. We take an opportunistic spectrum access approach, aiming to exploit unused spectrum while probabilistically bounding the interference to primary users. Let $\gamma_n \in (0, 1)$ be the *maximum allowed collision probability* with primary users on channel n, and $p_n^{tr}(t)$ the *transmission probability* on channel n for the base station in time slot t. The probability of collision caused by the base station should be kept below γ_n, i.e., $p_n^{tr}(t)\,[1 - a_n(t)] \leq \gamma_n$. In addition to primary user protection, another important objective is to exploit unused spectrum as much as possible. The transmission probability can be determined by jointly considering both objectives, as

$$p_n^{tr}(t) = \begin{cases} \min\left\{1, \frac{\gamma_n}{1 - a_n(t)}\right\}, & \text{if } 0 \leq a_n(t) < 1 \\ 1, & \text{if } a_n(t) = 1. \end{cases} \quad (6)$$

If $p_n^{tr}(t) = 1$, channel n will be accessed deterministically. If $p_n^{tr}(t) = \gamma_n/[1 - a_n(t)] < 1$, channel n will be accessed opportunistically with probability $p_n^{tr}(t)$.

D. Modulation-Coding Schemes

At the PHY layer, we consider various modulation and channel coding schemes. Without loss of generality, we assume several choices of modulation schemes, such as QPSK, 16-QAM and 64-QAM, combined with several choices of forward error correction (FEC) schemes, e.g., with rates 1/2, 2/3, and 3/4. We consider M unique combinations of modulation and FEC schemes, termed *Modulation-Coding* (MC) schemes, in this paper.

Under the same channel condition, different MC schemes will achieve different data rates and symbol error rates. Adaptive modulation and channel coding allow us to exploit user channel variations to maximize video data rate under a given residual bit error rate constraint. When a user has a good channel, it should adopt an MC scheme that can support a higher data rate. Conversely, it should adopt a low-rate MC scheme when the channel condition is poor.

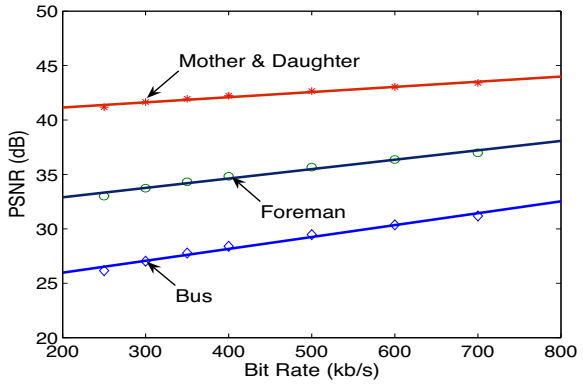


Fig. 3. Rate-PSNR curves with the MPEG-4 FGS codec for test sequences *Bus*, *Foreman*, and *Mother & Daughter*.

Let $\{MC_m\}_{m=1,\cdots,M}$ be the list of available MC schemes indexed according to their data rates in the increasing order. We assume slow fading channels with coherence time larger than a time slot. Each CR user measures its own channel and feedbacks measurements to the base station when its channel quality changes. At the beginning of a time slot, the base station is able to collect the number $n_{g,m}$ of users in each multicast group g who can successfully decode MC_m signals for $m = 1, 2, \cdots, M$.

E. Video Performance Measure

We consider the quality of reconstructed video (i.e., peak-signal-noise-ratio (PSNR) in dB) as performance measure. To address the heterogeneous user channels, each video g is encoded into one base layer with rate R_g^b and one enhancement layer with rate R_g^e. The total rate for video g is $R_g = R_g^b + R_g^e$. We adopt the FGS coding technique, such that the enhancement layer can be truncated at any bit location while *all* the remaining bits still being useful at the decoder [4]. The approximate "quality-rate" model used in [8] is adopted in our formulation:

$$Q_g(R_g) = Q_g^b + \beta_g(R_g - R_g^b) = Q_g^b + \beta_g R_g^e, \qquad (7)$$

where Q_g is the PSNR of video g, Q_g^b is the PSNR if only the base layer is received, and β_g is a constant depending on the specific video sequence and codec used. In Fig. 3, we plot the measured PSNRs (markers) and the estimated PSNRs using (7) (lines) for three test video sequences *Bus*, *Foreman*, and *Mother & Daughter*, where good matches can be observed.

Since the base layer carries the most important data, the most reliable MC scheme $MC_{b(g)}$ should be used, where $b(g) = \max_i\{i : n_{g,i} = N_g\}$, for all g. Without loss of generality, we assume that the base layer is always transmitted using MC_1. If a user's channel is so poor that it cannot decode the MC_1 signal, we consider it disconnected from the CR network. We further divide the enhancement layer into M sub-layers, where sub-layer m has rate $R_{g,m}^e$ and uses MC_m. Assuming that MC_m can carry $b_{g,m}$ bits of video g in one tile, we denote the number of tiles for sub-layer m of video g as $l_{g,m} \geq 0$. We have $R_g^e = \sum_{m=1}^{M} R_{g,m}^e = \sum_{m=1}^{M} b_{g,m} l_{g,m}$.

F. Proportional Fair Allocation

For data communications, *proportional fairness* is a widely adopted measure, which can be achieved by maximizing the sum of logarithms of user rates (i.e., utilities) [17]. Since we consider video quality in this paper, we define the utility for user i in group g as $U_{g,i} = \log Q_{g,i} = \log\left(Q_g^b + \beta_g R_g^e(i)\right)$, where $R_g^e(i)$ is the received enhancement layer rate of user i in group g.

The total utility for group g is $U_g = \sum_{i=1}^{N_g} U_{g,i}$. Intuitively, a lower layer should use a lower (i.e., more reliable) MC scheme. This is because if a lower layer is lost, a higher layer cannot be used at the decoder even if it is correctly received. Considering the user classification based on their MC schemes, we can rewrite U_g as follows [7]:

$$U_g = \sum_{k=1}^{M}(n_{g,k} - n_{g,k+1}) \log\left(Q_g^b + \beta_g \sum_{m=1}^{k} R_{g,m}^e\right), \qquad (8)$$

where $n_{g,M+1} = 0$. The utility function of the entire CR video multicast system is $U = \sum_{g=1}^{G} U_g$.

III. Optimized Video Multicast in CR Networks

A. Outline of the Proposed Approach

As discussed, the CR video multicast problem is highly challenging since many design choices are tightly coupled. First, as users see different channels, such heterogeneity should be accommodated so that a user can receive a video quality commensurate to its channel quality. Second, we need to determine the video rates before transmission, which, however, depend on future channel evolution and choice of MC schemes. Third, the trade-off between primary user protection and spectrum utilization should guide the scheduling of video packets to channels. Finally, all the optimization decisions should be made in real-time. Low-complexity, but efficient algorithms are needed, while theoretical optimality bounds would be highly appealing.

To address heterogeneous user channels, we adopt FGS to produce a base layer with rate R_g^b and an enhancement layer with rate $\bar{R}_g^e$. Without loss of generality, we assume R_g^b is prescribed for an acceptable video quality, while $\bar{R}_g^e$ is set to a large value that is allowed by the codec. During transmission, we determine the *effective rate* for each enhancement layer $R_g^e \leq \bar{R}_g^e$ depending on channel availability, sensing, and MC schemes.[2] The optimal partition of the enhancement layer should be determined such that each sub-layer uses a different MC scheme.

We determine the optimal partition of enhancement layers, the choices of MC schemes, and video packet scheduling as follows. First, we solve the optimal partition problem for every GoP based on an estimated (i.e., average) number of available tiles T_e in the next GoP window that can be used for the enhancement layer, using algorithm GRD1 with complexity $O(MGT_e)$. The tile allocations are then dynamically adjusted in each time slot according to more recent (and thus more accurate) channel status using algorithm GRD2, with complexity $O(MGK)$, where $K \ll T_e$. Second, during each time slot, video packets are scheduled to the available channels such that the overall system utility is maximized. The TSA algorithm

[2]The proposed approach can also be used for streaming stored FGS video.

has complexity $O(N \log N)$. Both GRD2 and TSA have low complexity and are suitable for execution in each time slot.

In real-time video, overdue packets generally do not contribute to improving the received quality. We assume that the data from a GoP should be be delivered in the next GoP window consisting of T_{GoP} time slots.[3] Since the base layer is essential for decoding a video, we assume that the base layers of all the videos are coded using MC_1. For the M sub-layers of the enhancement layer, a more important sub-layer will be coded using a more reliable (i.e., lower rate) MC scheme. At the beginning of each GoP window, all the base layers are transmitted using the available tiles. *Retransmissions* will be scheduled if no ACK is received for a base layer packet. After the base layers are transmitted, we allocate the remaining available tiles in the GoP window for the enhancement layer. The same rule applies to the enhancement sub-layers, such that a higher sub-layer will be transmitted if and only if all the lower sub-layers are acknowledged. This is due to the decoding dependency of layered video.

In each time slot t, the base station opportunistically access every channel n with probability $p_n^{tr}(t)$ given in (6). Specifically, for each channel n, the base station generates a random number $x_n(t)$, which is independent of the channel history $\theta_n(t)$ and uniformly distributed in [0,1]. If $x_n(t) \leq p_n^{tr}(t)$, the most important packet among those not ACKed in the previous GoP will be transmitted on channel n. If an ACK is received for this packet at the end of time slot t, this packet is successfully received by at least one of the users and will be removed from the transmission buffer. Otherwise, there is a collision with primary user and this packet will remain in the transmission buffer and will be retransmitted.

In the following, we describe in detail the three algorithms.

B. Enhancement Layer Partitioning and Tile Allocation

As a first step, we need to determine the effective rate for each enhancement layer $R_g^e \leq \bar{R}_g^e$. We also need to determine the optimal partition of each enhancement layer. Clearly, the solutions will be highly dependent on the channel availability processes and sensing results.

Recall that the base layers are transmitted using MC_1 first in each GoP window. The *remaining* available tiles can then be allocated to the enhancement layers. We assume that the number of tiles used for the enhancement layers in a GoP window, T_e, is known at the beginning of the GoP window. For example, we can estimate T_e by computing the total average "idle" intervals of all the N channels based on the channel model, decreased by the number of tiles used for the base layers (i.e., $R_g^b/b_{g,1}$). We then split the enhancement layer of each video g into M sub-layers, each occupying $l_{g,m}$ tiles when coded with MC_m, $m = 1, 2, \cdots, M$.

Letting $\vec{l} = [l_{1,1}, l_{1,2}, \cdots, l_{1,M}, l_{2,1}, \cdots l_{G,M}]$ denote the *tile allocation vector*, we formulate an optimization problem OPT-Part as follows.

$$\text{maximize: } U(\vec{l}) = \sum_{g=1}^{G} \sum_{k=1}^{M} (n_{g,k} - n_{g,k+1}) \times \log\left[Q_g^b + \beta_g \sum_{m=1}^{k} b_{g,m} l_{g,m}\right] \quad (9)$$

$$\text{subject to: } \sum_{g=1}^{G} \sum_{m=1}^{M} l_{g,m} \leq T_e \quad (10)$$

$$\sum_{m=1}^{M} b_{g,m} l_{g,m} \leq \bar{R}_g^e, \ g \in [1, \cdots, M] \quad (11)$$

$$l_{g,m} \geq 0, \ m \in [1, \cdots, M], g \in [1, \cdots, G]. \quad (12)$$

OPT-Part is solved at the beginning of each GoP window to determine the optimal partition of the enhancement layer. The objective is to maximize the overall system utility by choosing optimal values for the $l_{g,m}$'s. We can derive the effective video rates as $R_g^e = \sum_{m=1}^{M} b_{g,m} l_{g,m}$. The formulated problem is a MINLP problem, which is NP-hard [7]. In the following, we present two algorithms for computing near-optimal solutions to problem OPT-Part: (i) a *sequential fixing* (SF) algorithm based on a linear relaxation of (9), and (ii) a *greedy algorithm* GRD1 with proven optimality gap.

TABLE I
THE SEQUENTIAL FIXING (SF) ALGORITHM

1:	Use RLT to linearize the original problem
2:	Solved the LP relaxation
3:	Suppose $l_{\hat{g},\hat{m}}$ is the integer variable with the minimum $(\lceil l_{\hat{g},\hat{m}} \rceil - l_{\hat{g},\hat{m}})$ or $(l_{\hat{g},\hat{m}} - \lfloor l_{\hat{g},\hat{m}} \rfloor)$ value among all $l_{g,m}$ variables that remain to be fixed, round it up or down to the nearest integer
4:	If all $l_{g,m}$'s are fixed, got to Step 6
5:	Otherwise, reformulate a new relaxed LP with the newly fixed $l_{g,m}$ variables, and go to Step 2
6:	Output all fixed $l_{g,m}$ variables and $R_g^e = \sum_{m=1}^{M} b_{g,m} l_{g,m}$

1) A Sequential Fixing Algorithm : With this algorithm, the original MINLP is first linearized to obtain a linear programming (LP) relaxation. Then we iteratively solve the LP, while fixing one integer variable in every iteration [9], [10]. We use the *Reformulation-Linearization Technique* (RLT) to obtain the LP relaxation [18]. RLT is a technique that can be used to produce LP relaxations for a nonlinear, nonconvex polynomial programming problem. This relaxation will provide a tight upper bound for a maximization problem. Specifically, we linearize the logarithm function in (9) over some suitable, tightly-bounded interval using a polyhedral outer approximation comprised of a convex envelope in concert with several tangential supports. We further relax the integer constraints, i.e., allowing the $l_{g,m}$'s to take fractional values. Then we obtain an upper-bounding LP relaxation that can be solved in polynomial time. Due to lack of space, we refer interested readers to [18] for a detailed description of the technique.

We next solve the LP relaxation iteratively. During each iteration, we find the $l_{\hat{g},\hat{m}}$ which has the minimum value for $(\lceil l_{\hat{g},\hat{m}} \rceil - l_{\hat{g},\hat{m}})$ or $(l_{\hat{g},\hat{m}} - \lfloor l_{\hat{g},\hat{m}} \rfloor)$ among all fractional $l_{g,m}$'s, and round it up or down to the nearest integer. We next reformulate and solve a new LP with $l_{\hat{g},\hat{m}}$ fixed. This procedure repeats until all the $l_{g,m}$'s are fixed. The complete SF algorithm is given in Table I. The complexity of SF depends on the specific LP algorithm (e.g., the *simplex method* with polynomial-time average-case complexity).

2) A Greedy Algorithm : Although SF can compute a near-optimal solution in polynomial time, it does not provide any

[3]The proposed approach also works for the more general delay requirements that are multiple GoP windows.

TABLE II
THE GREEDY ALGORITHM (GRD1)

1: Initialize $l_{g,m}=0$ for all g and m
2: Initialize $A=\{1,2,\cdots,G\}$
3: WHILE $\left(\sum_{g=1}^{G}\sum_{m=1}^{M} l_{g,m} \le T_e$ and A is not empty$\right)$
4: Find $l_{\hat{g},\hat{m}}$ that can be increased by one:
$$\vec{e}_{\hat{g},\hat{m}} = \arg\max_{g\in A, m\in[1,\cdots,M]} \left\{ \frac{U(\vec{l}+\vec{e}_{g,m})-U(\vec{l})}{b_{g,m}+R/T_e} \right\}$$
5: $\vec{l}=\vec{l}+\vec{e}_{\hat{g},\hat{m}}$
6: IF $\left(\sum_m b_{\hat{g},m} l_{\hat{g},m} > \bar{R}_g^e\right)$
7: $\vec{l}=\vec{l}-\vec{e}_{\hat{g},\hat{m}}$
8: Delete $\hat{g}$ from A
9: END IF
10: END WHILE

guarantee on the optimality of the solution. In the following, we describe a greedy algorithm, termed GRD1, which exploits the inherent priority structure of layered video and MC schemes and has a proven optimality bound.

The complete greedy algorithm is given in Table II, where $R=\sum_{g=1}^{G}\bar{R}_g^e$ is the total rate of all the enhancement layers and $\vec{e}_i$ is a *unit vector* with "1" at the i-th location and "0" at all other locations. In GRD1, all the $l_{g,m}$'s are initially set to 0. During each iteration, one tile is allocated to the $\hat{m}$-th sub-layer of video $\hat{g}$. In Step 4, $l_{\hat{m},\hat{g}}$ is chosen to be the one that achieves the largest increase in terms of the "normalized" utility (i.e., $[U(\vec{l}+\vec{e}_{g,m})-U(\vec{l})]/[b_{g,m}+R/T_e]$) if it is assigned with an additional tile. Lines 6, 7, and 8 check if the assigned rate exceeds the maximum rate $\bar{R}_g^e$. GRD1 terminates when either all the available tiles are used or when all the video data are allocated with tiles. In the latter case, all the videos are transmitted at full rates. We have the following Theorem for GRD1. The proof is presented in the Appendix.

Theorem 1: The greedy algorithm GRD1 shown in Table II has a complexity $O(MGT_e)$. It guarantees a solution that is within a factor of $(1-e^{-1/2})$ of the global optimal solution.

3) A Refined Greedy Algorithm : GRD1 computes $l_{g,m}$'s based on an estimate of network status $\vec{S}(t)$ in the next T_{GoP} time slots. Due to channel dynamics, the computed $l_{g,m}$'s may not be exactly accurate, especially when T_{GoP} is large. We next present a refined greedy algorithm, termed GRD2, which adjusts the $l_{g,m}$'s based on more accurate estimation of the channel status.

GRD2 is executed at the beginning of every time slot. It estimates the number of available tiles $T_e(t)$ in the next T_{est} successive time slots, where $1 \le T_{est} \le T_{GoP}$ is a design parameter depending on the coherence time of the channels. Such estimates are more accurate than that in GRD1 since they are based on recently received ACKs and recent sensing results. Specifically, we estimate $T_e(t)$ using the belief vector $\vec{a}(t)$ in time slot t. Recall that $a_n(t)$'s are computed based on the channel model, feedback, sensing results, and sensing errors, as given in (3), (4), and (5). For the next time slot, $a_n(t+1)$ can be estimated as $\hat{a}_n(t+1)=\lambda_n a_n(t)+\mu_n[1-a_n(t)]=(\lambda_n-\mu_n)a_n(t)+\mu_n$. Recursively, we can derive $\hat{a}_n(t+\tau)$ for the next τ time slots.

$$\hat{a}_n(t+\tau)=(\lambda_n-\mu_n)^{\tau}a_n(t)+\mu_n\frac{1-(\lambda_n-\mu_n)^{\tau}}{1-(\lambda_n-\mu_n)}. \quad (13)$$

TABLE III
THE REFINED GREEDY ALGORITHM (GRD2) FOR EACH TIME SLOT

1: Initialize $l_{g,m}=0$ for all g and m
2: Initialize $A=\{1,2,\cdots,G\}$
3: Initialize $N_{ack}(0)=0$
4: Estimate $T_e(1)$ based on the Markov Chain channel model
5: Use GRD1 to find all $l_{g,m}$'s based on $T_e(1)$
6: WHILE $t=2$ to T_{GoP}
7: Estimate $T_e(t)$
8: IF $[T_e(t)+N_{ack}(t-1) < T_e(t-1)+N_{ack}(t-2)]$
9: WHILE $\left[\sum_{g=1}^{G}\sum_{m=1}^{M} l_{g,m} > T_e(t)+N_{ack}(t-2)\right]$
10: Find $l_{\hat{g},\hat{m}}$ that can be reduced by 1:
$$\vec{e}_{\hat{g},\hat{m}} = \arg\min_{\forall g,m\in\{m',\cdots,M\}} \left\{ \frac{U(\vec{l})-U(\vec{l}-\vec{e}_{g,m})}{b_{g,m}+R/T_e} \right\}$$
11: $\vec{l}=\vec{l}-\vec{e}_{\hat{g},\hat{m}}$
12: IF $(\hat{g}\notin A)$
13: Add $\hat{g}$ to A
14: END IF
15: END WHILE
16: END IF
17: IF $[T_e(t)+N_{ack}(t-1) > T_e(t-1)+N_{ack}(t-2)]$
18: WHILE $\left[\sum_{g=1}^{G}\sum_{m=1}^{M} l_{g,m} \le T_e(t)+N_{ack}(t-1)$ and A is not empty$\right]$
19: Find $l_{\hat{g},\hat{m}}$ that can be increased by 1
$$\vec{e}_{\hat{g},\hat{m}} = \arg\max_{g\in A, m\in\{m',\cdots,M\}} \left\{ \frac{U(\vec{l}+\vec{e}_{g,m})-U(\vec{l})}{b_{g,m}+R/T_e} \right\}$$
20: $\vec{l}=\vec{l}+\vec{e}_{\hat{g},\hat{m}}$
21: IF $\left(\sum_m b_{\hat{g},m} l_{\hat{g},m} > \bar{R}_g^e\right)$
22: $\vec{l}=\vec{l}-\vec{e}_{\hat{g},\hat{m}}$
23: Delete $\hat{g}$ from A
24: END IF
25: END WHILE
26: END IF
27: Update $N_{ack}(t-1)$
28: END WHILE

At the beginning section of a GoP window, all the base layers will be firstly transmitted. We start the estimation after all the base layers have been successfully received (possibly with retransmissions). The number of available tiles in the following T_{est} time slots can be estimated as $T_e(t)=\sum_{n=1}^{N}\sum_{\tau=0}^{t_{min}}\hat{a}_n(t+\tau)$, where $\hat{a}_n(t+0)=a_n(t)$ and $t_{min}=\min\{T_{est}-1, T_{GoP}-(t \bmod T_{GoP})\}$. $T_e(t)$ may not be an integer, but it does not affect the outcome of GRD2.

We then adjust the $l_{g,m}$'s based on $T_e(t)$ and $N_{ack}(t)$, the number of ACKs received in time slot t. If $T_e(t)+N_{ack}(t-1) > T_e(t-1)+N_{ack}(t-2)$, there are more tiles that can be allocated and we can increase some of the $l_{g,m}$'s. On the other hand, if $T_e(t)+N_{ack}(t-1) < T_e(t-1)+N_{ack}(t-2)$, we have to reduce some of the $l_{g,m}$'s. Due to layered videos, when we increase the number of allocated tiles, we only need to consider $l_{g,m}$ for $m=m', m'+1,\cdots,M$, where $MC_{m'}$ is the highest MC scheme used in the previous time slot. Similarly, when we reduce the number of allocated tiles, we only need to consider $l_{g,m}$ for $m=m', m'+1,\cdots,M$.

The refined greedy algorithm is given in Table III. For time slot t, the complexity of GRD2 is $O(MGK)$, where $K=|N_{ack}(t-1)-N_{ack}(t-2)+T_e(t)-T_e(t-1)|$. Since $K \ll T_e$, the complexity of GRD2 is much lower than GRD1, suitable for execution in each time slot.

C. Tile Scheduling in a Time Slot

In each time slot t, we need to schedule the remaining tiles for transmission on the N channels. We define $\text{Inc}(g,m,i)$ to

TABLE IV
ALGORITHM FOR TILE SCHEDULING IN A TIME SLOT

1:	Initialize m_g to the lowest MC that has not been ACKed for all g
2:	Initialize i_g to the first packet that has not been ACKed for all g
3:	Sort $\{c_n(t)\}$ in decreasing order. Let the sorted channel list be indexed by j.
4:	While ($j = 1$ to N)
5:	Find the group having the maximum increase in $U(g)$: $\hat{g} = \arg\max_{\forall g} \text{Inc}(g, m_g, i_g)$
6:	Allocate the tile on channel j to group $\hat{g}$
7:	Update $m_{\hat{g}}$ and $i_{\hat{g}}$
8:	End while

be the increase in the group utility function $U(g)$ after the i-th tile in the sub-layer using MC_m is successfully decoded. It can be shown that

$$\text{Inc}(g,m,i) = \sum_{k=m}^{M}(n_{g,k} - n_{g,k+1}) \times \log\left[1 + \frac{\beta_g b_{g,m}}{Q_g^b + \beta_g \sum_{u=1}^{m-1} b_{g,u} l_{g,u} + (i-1)\beta_g b_{g,m}}\right].$$

$\text{Inc}(g,m,i)$ can be interpreted as the *reward* if the tile is successfully received.

Letting $c_n(t)$ be the probability that the tile is successfully received, we have $c_n(t) = p_n^{tr}(t)a_n(t)$. Our objective of tile scheduling is to maximize the expected reward, i.e.,

$$\text{maximize:} \quad \text{E[Reward}(\vec{\xi})] = \sum_{n=1}^{N} c_n(t) \cdot \text{Inc}(\xi_n), \qquad (14)$$

where $\vec{\xi} = \{\xi_n\}_{n=1,\cdots,N}$ and ξ_n is the tile allocation for channel n, i.e., representing the three-tuple $\{g,m,i\}$. The TSA algorithm is shown in Table IV, which solves the above optimization problem. The complexity of TSA is $O(N \log N)$. We have the following theorem for TSA.

Theorem 2: E[Reward] is maximized if $\text{Inc}(\xi_i) > \text{Inc}(\xi_j)$ when $c_i(t) > c_j(t)$ for all i and j.

Proof: Suppose there exists a pair of i and j where $\text{Inc}(\xi_i) > \text{Inc}(\xi_j)$ and $c_i(t) < c_j(t)$. We can further increase E[Reward] by switching the tile assignment, i.e., assign channel i to ξ_j and channel j to ξ_i. With this new assignment, the net increase in E[Reward] is

$$\begin{aligned} & c_j(t)\text{Inc}(\xi_i) + c_i(t)\text{Inc}(\xi_j) - c_i(t)\text{Inc}(\xi_i) - c_j(t)\text{Inc}(\xi_j) \\ = \;& [c_j(t) - c_i(t)][\text{Inc}(\xi_i) - \text{Inc}(\xi_j)] > 0. \end{aligned}$$

Therefore E[Reward] is maximized when the $\{\text{Inc}(\xi_i)\}$ and $\{c_i(t)\}$ are in the same order. ■

IV. SIMULATION RESULTS

We evaluate the performance of the proposed CR video multicast framework using a customized simulator implemented with a combination of C and MATLAB. Specifically, the LPs are solved using the MATLAB Optimization Toolbox and the remaining parts are written in C. For the results reported in this section, we have $N = 12$ channels (unless otherwise specified). The channel parameters λ_n and μ_n are set between $(0,1)$. The maximum allowed collision probability γ_n is set to 0.2 for all the channels unless otherwise specified.

The CR base station multicasts three Common Intermediate Format (CIF, 352×288) video sequences to three multicast groups, i.e., *Bus* to group 1, *Foreman* to group 2, and *Mother & Daughter* to group 3. The $n_{1,m}$'s are $\{42, 40, 36, 30, 22, 12\}$ (i.e., 42 users can decode MC_1 signal, 40 users can decode MC_2 signal, and so forth); the $n_{2,m}$'s are $\{51, 46, 40, 32, 23, 12\}$ and the $n_{3,m}$'s are $\{49, 44, 40, 32, 24, 13\}$. The number of bits carried in one tile using the MC schemes are 1 kb/s, 1.5 kb/s, 2 kb/s, 3 kb/s, 5.3 kb/s, and 6 kb/s, respectively. We choose T_{GoP}=150 and $T_{est} = 10$, sensing interval $W = 3$, false alarm probability $\epsilon_n = 0.3$ and miss detection probability $\delta_n = 0.25$ for all n, unless otherwise specified.

In every simulation, we compare three schemes: (i) a simple heuristic scheme that equally allocates tiles to each group (Equal Allocation); (ii) A scheme based on SF (Sequential Fixing), and (iii) a scheme based on the greedy algorithm GRD2 (Greedy Algorithm). These schemes have increasing complexity in the order of Equal Allocation, Greedy Algorithm, and Sequential Fixing. They differ on how to solve Problem OPT-Part, while the same tile scheduling algorithm and opportunistic spectrum access scheme are used in all the schemes. Each point in the figures is the average of 10 simulation runs, with 95% confidence intervals plotted. We observe that the 95% confidence intervals for Equal Allocation and Greedy Algorithm are negligible, while the 95% confidence intervals for Sequential Fixing is relatively larger. The C/MATLAB code is executed in a Dell Precision Workstation 390 with an Intel Core 2 Duo E6300 CPU working at 1.86 GHz and a 1066 MB memory. For number of channels ranging from N=3 to N=15, the execution times of Equal Allocation and Greedy Algorithm are about a few milliseconds, while Sequential Fixing takes about two seconds.

In Fig. 4 we plot the average PSNR among all users in each multicast group. For all the groups, Greedy Algorithm achieves the best performance, with up to 4.2 dB improvements over Equal Allocation and up to 0.6 dB improvements over Sequential Fixing. We find Sequential Fixing achieves a lower PSNR than Equal Allocation for group 3, but higher PSNRs for groups 1 and 2. This is because Equal Allocation does not consider channel conditions and fairness. It achieves better performance for group 3 at the cost of much lower PSNRs for groups 1 and 2. We also plot Frame 53 from the original *Bus* sequence and the decoded video at user 1 of group 1 in Fig 5. We choose this user since it is one of the users with lowest PSNR values. The average PSNR of this user is 29.54 dB, while the average PSNR of all group 1 users is 34.6 dB. Compared to the original frame (right), the reconstructed frame (left) looks quite good, although some details are lost.

In Fig. 6, we examine the impact of the maximum allowed collision probability γ_n. We increase γ_n from 0.1 to 0.3, and plot the average PSNR values among all the users. When γ_n gets larger, there will be higher chance of collision for the video packets, which hurts the received video quality. However, a higher γ_n also allows a higher transmission probability $p_n^{tr}(t)$ for the base station (see (6)), thus allowing the base station to grab more spectrum opportunities and achieve a higher video rate. The net effect of these two contradicting effects is improved video quality for the range of γ_n values considered in this simulation. This is illustrated in the figure where all the three curves increase as γ_n gets larger. We also observe that the curves for Sequential Fixing and Equal Allocation are roughly parallel to each other, while the Greedy Algorithm curve has a steeper slope. This indicates

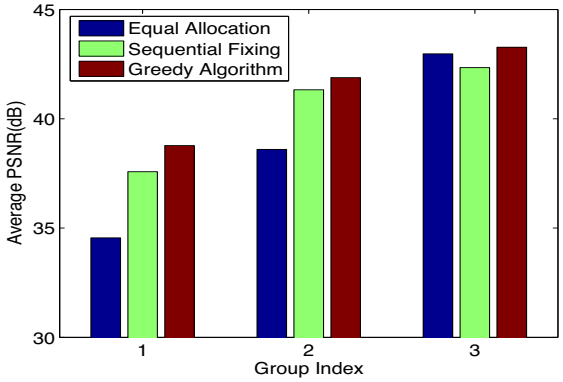


Fig. 4. Average PSNR of all multicast users.

Fig. 5. The decoded Frame 53 (the left one) and the original Frame 53 (the right one) at user 1 in group 1.

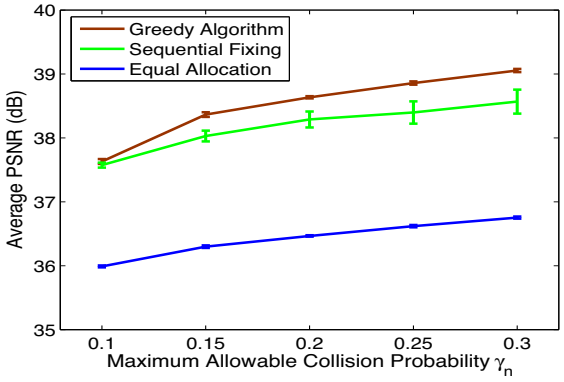


Fig. 6. Average PSNR of all users versus γ_n (with 95% confidence intervals).

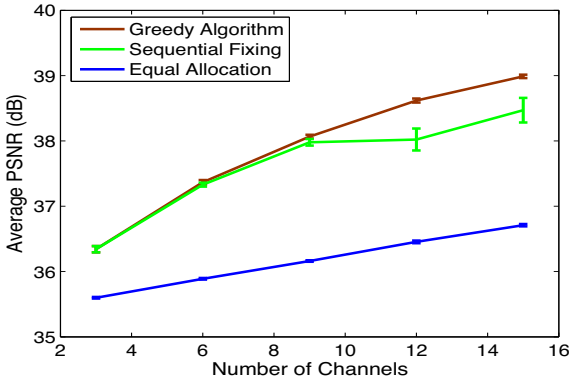


Fig. 7. Average PSNR of all users versus N (with 95% confidence intervals).

that Greedy Algorithm is more efficient in exploiting the additional bandwidth allowed by an increased γ_n.

In Fig. 7, we examine the impact of number of channels N. We increase N from 3 to 15 in steps of 3, and plot the average PSNR values of all multicast users. As expected, the more channels, the more spectrum opportunities for the CR networks, and the better the video quality. Again, we observe that the Greedy Algorithm curve has the steepest slope, implying it is more efficient in exploiting the increased spectrum opportunity for video transmissions.

We demonstrate the impact of sensing errors in Fig. 8. We test five sets of $\{\epsilon_n, \delta_n\}$ values as follows: $\{0.10, 0.38\}$, $\{0.30, 0.25\}$, $\{0.5, 0.17\}$, $\{0.70, 0.10\}$ and $\{0.9, 0.04\}$ [11], and plot the average PSNR values of all users. It is quite interesting to see that the video quality is not very sensitive to sensing errors. Even as ϵ_n is increased nine times from 10% to 90%, there is only 0.58 dB reduction (or a 1.5% normalized reduction) in average PSNR when Greedy Algorithm is used. The same can be observed for the other two curves. We conjecture that this is due to the opportunistic spectrum access approach adopted in all the three schemes. A special strength of the proposed approach is that it explicitly considers both types of sensing errors and mitigates the impact of both sensing errors. For example, when the false alarm rate is very high, the base station will not trust the sensing results and will access the channel relatively more aggressively, thus mitigating the negative effect of the high false alarm rate.

Finally, we demonstrate the impact of user channel variations (i.e., due to mobility). We chose a tagged user in group 1 and assume that its channel condition changes every 20 GoPs. The highest MC scheme that the tagged user can decode is changed according to the following sequence: MC3, MC5, MC4, MC6, MC5 and MC3. All other parameters remain the same as in the previous experiments. In Fig. 9, we plot the average PSNRs for each GoP at this user that are obtained using the three algorithms. We observe that both Greedy Algorithm and Sequential Fixing can quickly adapt to changing channel conditions. Both algorithms achieve received video qualities commensurate with the channel quality of the tagged user. We also find the video quality achieved by Greedy Algorithm is more stable than that of Sequential Fixing, while the latter curve has some deep fades from time to time. This is due to the fact that Greedy Algorithm has a proven optimality bound, while Sequential Fixing does not provide any guarantee. The Equal Allocation curve is relative constant for the entire period since it does not adapt to channel variations. Although being simple, it does not provide good video quality in this case.

For optimization-driven multimedia systems, there is a trade-off between (i) grabbing all the available resource to maximize media quality and (ii) be less adaptive to network dynamics for a smooth playout. The main objective of this paper is to demonstrate the feasibility and layout the framework for video streaming over infrastructure-based CR networks, using an objective function of maximizing the overall user utility. We will investigate the interesting problem of trading off resource utilization and smoothness in our future work.

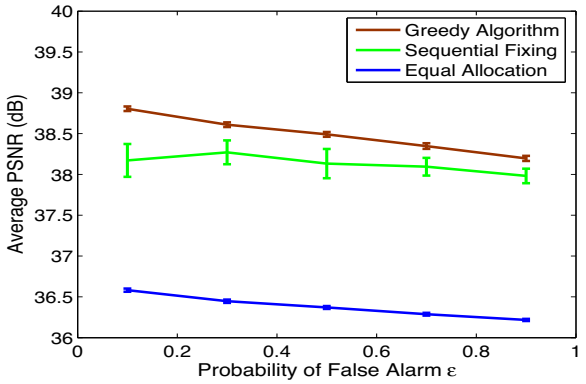


Fig. 8. Average PSNR of all users for various $\{\epsilon_n, \delta_n\}$ values (with 95% confidence intervals).

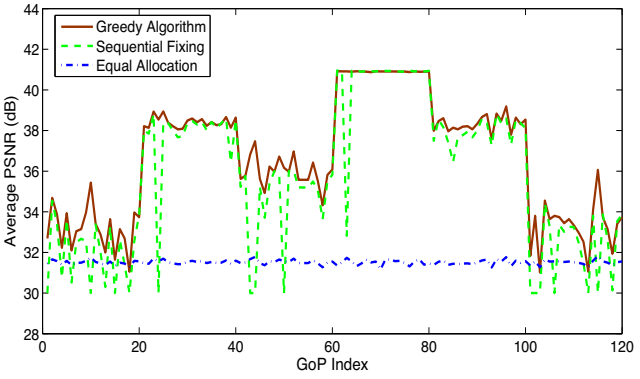


Fig. 9. GoP average PSNRs of a tagged user in Group 1, when its channel condition varies over time.

V. Related Work

As observed [1], [2], the mainstream CR research has been focused on spectrum sensing and dynamic spectrum access issues. For example, the approach of iteratively sensing a selected subset of available channels has been adopted in the design of CR MAC protocols (e.g., see [12]). The important trade-off between the two types of sensing errors is addressed in depth in [11].

The equally important QoS issue has been considered only in a few papers [19]–[25], where the focus is on the so-called "network-centric" metrics such as maximum throughput and delay [22]. In [25], a game-theoretic framework is described for resource allocation for multimedia transmissions in spectrum agile wireless networks. In this interesting work, each wireless station plays a resource management game, which is coordinated by a network moderator. A mechanism-based resource management scheme determines the amount of transmission time to be allocated to various users on different frequency bands such that certain global system metrics are optimized.

Video multicast, as one of the most important multimedia services, has attracted considerable efforts from the research community. Layered video multicast has been studied in the context of mobile ad hoc networks (e.g., see [5], [6]) and infrastructure-based wireless networks (e.g., see [7], [8]). A greedy algorithm is proposed in [7] for layered video multicast in WiMAX networks with a proven optimality gap $(1-e^{-1/2})$. The proposed GRD1 algorithm extends the work in [7] for FGS video under dynamic channel availability processes. The main difference between this and the prior studies is that, unlike the prior work where the spectrum is exclusively used by video sessions, we need to consider the presence and protection of primary users, which makes the problem more interesting and challenging.

VI. Conclusion

In this paper, we addressed the problem of multicasting FGS video in CR networks. The problem formulation took video quality and proportional fairness as objectives, while considering cross-layer design factors such as FGS coding, spectrum sensing, opportunistic spectrum access, primary user protection, scheduling, error control and modulation. We proposed efficient optimization and scheduling algorithms for highly competitive solutions, and proved the complexity and optimality bound of the proposed greedy algorithm. Our simulation results demonstrate not only the viability of video over CR networks, but also the efficacy of the proposed approach.

Appendix
Proof of Theorem 1

Proof: (i) *Complexity*: In Step 4 in Table II, it takes $O(MG)$ to solve for $\vec{e}_{\hat{g},\hat{m}}$. Since each iteration assigns one tile to sub-layer $\hat{m}$ of group $\hat{g}$, it takes T_e iterations to allocate all the available tiles in a GoP window. Therefore, the overall complexity of GRD1 is $O(MGT_e)$.

(ii) *Optimality Bound*: This proof is extended from a result first shown in [7] for layered videos. We first show a property of group utility $U_g(\vec{l})$, which will be used in the proof of the optimality gap. For two vectors $\vec{l}_g^1$ and $\vec{l}_g^2$, letting $\Delta = U_g(\vec{l}_g^1) - U_g(\vec{l}_g^2)$, we have

$$\begin{aligned}
\Delta &= \textstyle\sum_{k=1}^{M}(n_{g,k} - n_{g,k+1}) \times \\
&\qquad \log\left(1 + \frac{\sum_{m=1}^{k} \beta_g b_{g,m}(l_{g,m}^1 - l_{g,m}^2)}{Q_g^b + \sum_{m=1}^{k} \beta_g b_{g,m} l_{g,m}^2}\right) \\
&\le \textstyle\sum_{k=1}^{M}\sum_{m=1}^{k}(l_{g,m}^1 - l_{g,m}^2)^+(n_{g,k} - n_{g,k+1}) \times \\
&\qquad \log\left(1 + \beta_g b_{g,m} / \left[Q_g^b + \textstyle\sum_{m=1}^{k}\beta_g b_{g,m} l_{g,m}^2\right]\right) \\
&\le \textstyle\sum_{k=1}^{M}\sum_{m=1}^{M}(l_{g,m}^1 - l_{g,m}^2)^+(n_{g,k} - n_{g,k+1}) \times \\
&\qquad \log\left(1 + \beta_g b_{g,m} / \left[Q_g^b + \textstyle\sum_{m=1}^{k}\beta_g b_{g,m} l_{g,m}^2\right]\right) \\
&= \textstyle\sum_{m=1}^{M}(l_{g,m}^1 - l_{g,m}^2)^+ \left[U_g(\vec{l}_g^2 + b_{g,m}) - U(\vec{l}_g^2)\right], \quad (15)
\end{aligned}$$

where $y^+ = \max\{0, y\}$. The first inequality is due to the concavity of logarithm functions.

Next we prove the optimality bound. Let $\vec{l}_t$ be the output of GRD1 after t iterations. Let the utility gap between the optimal solution and the GRD1 solution be $F_t = U(\vec{l}^*) - U(\vec{l}_t)$, and $\vec{e}_{\hat{g},\hat{m}}(t)$ the argument found in Step 4 of GRD1

after t iterations. We have $\vec{l}_t = \vec{l}_{t-1} + \vec{e}_{\hat{g},\hat{m}}(t)$ and

$$\begin{aligned} & F_{t-1} = U(\vec{l}^*) - U(\vec{l}_{t-1}) \\ \leq & \textstyle\sum_g \sum_m (l^*_{g,m} - l_{g,m})^+ [U(\vec{l}_{t-1} + \vec{e}_{g,m}(t)) - U(\vec{l}_{t-1})] \\ \leq & \textstyle\sum_g \sum_m (l^*_{g,m} - l_{g,m})^+ \\ & \qquad [U(\vec{l}_{t-1} + \vec{e}_{\hat{g},\hat{m}}(t)) - U(\vec{l}_{t-1})] \frac{b_{g,m} + R/T_e}{b_{\hat{g},\hat{m}}(t) + R/T_e} \\ \leq & \frac{U(\vec{l}_t) - U(\vec{l}_{t-1})}{b_{\hat{g},\hat{m}}(t) + R/T_e} \textstyle\sum_g \sum_m [l^*_{g,m}(b_{g,m} + R/T_e)]. \end{aligned}$$

The first inequality is due to (15) and the second inequality follows Step 4 of GRD1. It follows (10) that $\sum_g \sum_m l^*_{g,m} \leq T_e$ and $\sum_g \sum_m b_{g,m} l^*_{g,m} \leq R$. We have $F_{t-1} \leq (F_{t-1} - F_t) \frac{2R}{b_{\hat{g},\hat{m}}(t) + R/T_e}$. Solving for F_t, we have $F_t \leq F_{t-1} \{1 - [b_{\hat{g},\hat{m}}(t) + R/T_e]/(2R)\}$.

Suppose the *WHILE* loop in Table II has been executed k times when the solution is obtained.

$$\begin{aligned} F_k & \leq F_{k-1} \{1 - [b_{\hat{g},\hat{m}}(k) + R/T_e]/(2R)\} \\ & \leq F_0 \textstyle\prod_{t=1}^k \{1 - [b_{\hat{g},\hat{m}}(t) + R/T_e]/(2R)\} \\ & \leq F_0 \left\{1 - 1/(2kR) \textstyle\sum_{t=1}^k [b_{\hat{g},\hat{m}}(t) + R/T_e]\right\}^k. \end{aligned}$$

The *WHILE* loop exits when one or both of two constraints are violated. If $\sum_g \sum_m l_{g,m} \leq T_e$ is violated, there is no tile that can be used. Therefore $k \geq T_e$ and $\sum_{t=1}^k R/T_e \geq R$. If constraint "$A$ is not empty" is violated, all the videos have been allocated sufficient number of tiles and will be transmitted at full rates. We have $\sum_{t=1}^k b_{\hat{g},\hat{m}}(t) \geq R$ in this case. It follows that

$$\begin{aligned} F_k & \leq F_0 \left\{1 - 1/(2kR) \textstyle\sum_{t=1}^k [b_{\hat{g},\hat{m}}(t) + R/T_e]\right\}^k \\ & \leq F_0 \left[1 - 1/(2k)\right]^k \leq F_0 e^{-1/2}. \end{aligned}$$

Since $F_0 = U(\vec{l}^*)$, we have $U(\vec{l}_k) \geq (1 - e^{-1/2}) U(\vec{l}^*)$. Therefore, we conclude that the GRD1 solution is bounded by $(1 - e^{-1/2}) U(\vec{l}^*)$ and $U(\vec{l}^*)$. ■

REFERENCES

[1] I. Akyildiz, W. Lee, M. Vuran, and S. Mohanty, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey," *Computer Netw. J.*, vol. 50, no. 9, pp. 2127–2159, Sept. 2006.

[2] Q. Zhao and B. Sadler, "A survey of dynamic spectrum access," *IEEE Signal Process. Mag.*, vol. 24, no. 3, pp. 79–89, May 2007.

[3] Y. Zhao, S. Mao, J. Neel, and J. Reed, "Performance evaluation of cognitive radios: metrics, utility functions, and methodologies," *Proc. IEEE*, Special Issue on Cognitive Radio, vol. 97, no. 4, pp.642–659, Apr. 2009.

[4] H. Radha, M. van der Schaar, and Y. Chen, "The MPEG-4 fine-grained scalable video coding method for multimedia streaming over IP," *IEEE Trans. Multimedia*, vol. 3, no. 1, pp. 53–68, Mar. 2001.

[5] S. Mao, X. Cheng, Y.T. Hou, and H. Sherali, "Multiple description video multicast in wireless ad hoc networks," *ACM/Kluwer Mobile Netw. Appl. J.*, vol. 11, no. 1, pp. 63–73, January 2006.

[6] W. Wei and A. Zakhor, "Multiple tree video multicast over wireless ad hoc networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 1, pp. 2–15, Jan. 2007.

[7] S. Deb, S. Jaiswal, and K. Nagaraj, "Real-time video multicast in WiMAX networks," in *Proc. IEEE INFOCOM'08*, Phoenix, AZ, Apr. 2008, pp. 1579–1587.

[8] M. van der Schaar, S. Krishnamachari, S. Choi, and X. Xu, "Adaptive cross-layer protection strategies for robust scalable video transmission over 802.11 WLANs," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 10, pp. 1752–1763, Dec. 2003.

[9] Y.T. Hou, Y. Shi, and H.D. Sherali, "Spectrum sharing for multi-hop networking with cognitive radios," *IEEE Journal on Selected Areas in Communications,* vol. 26, issue 1, pp. 146–155, Jan. 2008.

[10] Y.T. Hou, Y. Shi, and H.D. Sherali, "Optimal base station selection for anycast routing in wireless sensor networks," *IEEE Transactions on Vehicular Technology,* vol. 55, issue 3, pp. 813–821, May 2006.

[11] Y. Chen, Q. Zhao, and A. Swami, "Joint design and separation principle for opportunistic spectrum access in the presence of sensing errors," *IEEE Trans. Inf. Theory*, vol. 54, no. 5, pp. 2053–2071, May 2008.

[12] Q. Zhao, S. Geirhofer, L. Tong, and B. Sadler, "Opportunistic spectrum access via periodic channel sensing," *IEEE Trans. Signal Process.*, vol. 36, no. 2, pp. 785–796, Feb. 2008.

[13] S. Geirhofer, L. Tong, and B. Sadler, "Cognitive medium access: constraining interference based on experimental models," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 1, pp. 95–105, Jan. 2008.

[14] J. Nonnenmacher and E. Biersack, "Optimal multicast feedback," in *Proc. IEEE INFOCOM'98*, San Francisco, CA, Mar./Apr. 1998, pp. 964–971.

[15] U. Berthold and F. Jondral, "Guidelines for designing OFDM overlay systems," in *Proc. IEEE DySPAN'05*, Baltimore, MD, Nov. 2005, pp. 626–629.

[16] H. Tang, "Some physical layer issues of wide-band cognitive radio systems," in *Proc. IEEE DySPAN'05*, Baltimore, MD, Nov. 2005, pp. 151–159.

[17] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *J. Operational Research Society*, vol. 49, no. 3, pp. 237–252, Mar. 1998.

[18] S. Kompella, S. Mao, Y. Hou, and H. Sherali, "On path selection and rate allocation for video in wireless mesh networks," *IEEE/ACM Trans. Netw.*, vol. 17, no. 1, pp. 212–224, Feb. 2009.

[19] L.-C. Wang, A. Chen, and D. Wei, "A cognitive MAC protocol for QoS provisioning in overlaying ad hoc networks," in *Proc. IEEE CCNC'07*, Las Vegas, NV, Jan. 2007, pp. 1139–1143.

[20] S. Srinivasa and S. Jafar, "The throughput potential of cognitive radio: A theoretical perspective," *IEEE Commun. Mag.*, vol. 45, no. 5, pp. 73–79, May 2007.

[21] T. Weingart, D. Sicker, and D. Grunwald, "A statistical method for reconfiguration of cognitive radios," *IEEE Wireless Commun. Mag.*, vol. 14, no. 4, pp. 34–40, Aug. 2007.

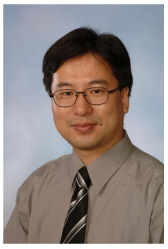
[22] H. Su and X. Zhang, "Cross-layer based opportunistic MAC protocols for QoS provisioning over cognitive radio wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 1, pp. 118–129, Jan. 2008.

[23] S. Jafar and S. Srinivasa, "Capacity limits of cognitive radio with distributed and dynamic spectral activity," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 3, pp. 529–537, Apr. 2007.

[24] C.-T. Chou, S. S. N., H. Kim, and K. Shin, "What and how much to gain by spectral agility," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 3, pp. 576–588, Apr. 2007.

[25] A. Fattahi, F. Fu, M. van der Schaar, and F. Paganni, "Mechanism-based resource allocation for multimedia transmission over spectrum agile wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 3, pp. 601–612, Apr. 2007.

Donglin Hu received the M.S. degree from Tsinghua University, Beijing, China, in 2007 and the B.S. degree from Nanjing University of P&T, Nanjing, China in 2004, respectively, all in electrical engineering. Since 2007, he has been pursuing the Ph.D. degree in the Department of Electrical and Computer Engineering, Auburn University, Auburn, AL. His research interests include cognitive radio networks, cross-layer optimization, algorithm design for wireless network and multimedia communications.



Shiwen Mao (S'99-M'04-SM'09) received his Ph.D. degree in electrical and computer engineering from Polytechnic Institute of New York University, Brooklyn, NY (formerly known as Polytechnic University) in 2004. He was a Research Member with IBM China Research Lab, Beijing, P.R. China from 1997 to 1998 and a Research Intern with Avaya Labs-Research, Holmdel, NJ in summer 2001. He was a Research Scientist in the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA from 2003 to 2006. Currently, he is an Assistant Professor in the Department of Electrical and Computer Engineering, Auburn University, Auburn, AL.

Dr. Mao's research interests include cross-layer optimization of wireless networks and multimedia communications, with current focus on cognitive radio networks and free space optical networks. He is on the Editorial Board of Elsevier Ad Hoc Networks Journal, Wiley International Journal of Communication Systems, and ICST Transactions on Mobile Communications and Applications. He is a coauthor of *TCP/IP Essentials: A Lab-Based Approach* (Cambridge, U.K.: Cambridge University Press, 2004). Dr. Mao received the *2004 IEEE Communications Society Leonard G. Abraham Prize in the Field of Communications Systems* and the *Best Paper Runner-up Award* from the Fifth International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine) 2008. He holds one US patent.

Y. Thomas Hou (S'91–M'98–SM'04) received his B.E. degree from the City College of New York in 1991, M.S. degree from Columbia University in 1993, and Ph.D. degree from Polytechnic Institute of New York University (formerly known as Polytechnic University) in 1998, all in Electrical Engineering. From 1997 to 2002, Dr. Hou was a Researcher at Fujitsu Laboratories of America, Sunnyvale, CA. Since 2002, he has been with Virginia Polytechnic Institute and State University ("Virginia Tech"), the Bradley Department of Electrical and Computer Engineering, Blacksburg, VA, where he is now an Associate Professor.

Prof. Hou's research interests are cross-layer design and optimization for MIMO-based ad hoc networks, cognitive radio wireless networks, cooperative communications, video communications over dynamic ad hoc networks, and algorithm design for sensor networks. He is a recipient of an Office of Naval Research (ONR) Young Investigator Award (2003) and a National Science Foundation (NSF) CAREER Award (2004) for his research on optimizations and algorithm design for wireless ad hoc and sensor networks. He has published extensively in leading IEEE and ACM journals and top-tier IEEE and ACM conferences and received five best paper awards from IEEE (including IEEE INFOCOM 2008 Best Paper Award and IEEE ICNP 2002 Best Paper Award). He holds five U.S patents.

Prof. Hou is currently serving as an Area Editor of IEEE Transactions on Wireless Communications and an Associate Editor of IEEE Transactions on Mobile Computing, IEEE Wireless Communications Magazine, ACM/Springer Wireless Networks (WINET), and Elsevier Ad Hoc Networks. He was a past Associate Editor of IEEE Transactions on Vehicular Technology. He was Co-Chair of Technical Program Committee for IEEE INFOCOM 2009, which was held in Rio de Janeiro, Brazil.

Jeffrey H. Reed (S'78–M'80–SM'98–F'05) is the Willis G. Worcester Professor in the Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg, and Chief Technical Officer, Cognitive Radio Technologies, Lynchburg. From June 2000 to June 2002, he was Director of the Mobile and Portable Radio Research Group. He is currently Director of the newly formed umbrella wireless organization Wireless@Virginia Tech.

His area of expertise is in software radios, smart antennas, wireless networks, and communications signal processing. He is the author of *Software Radio: A Modern Approach to Radio Design* (Englewood Cliffs, NJ: Prentice-Hall, 2002). His latest book is *An Introduction to Ultra Wideband Communication Systems* (Englewood Cliffs, NJ: Prentice-Hall, 2005), considered one of the most comprehensive books on ultra-wideband communications.

Dr. Reed received the College of Engineering Award for Excellence in Research in 2001. In 2004 he received an award from the SDR Forum for his pioneering 2002 publication that provides a mathematical foundation to cognitive radio based on game theory.