Progressive Pseudo-analog Transmission for Mobile Video Streaming

Dongliang He, Cuiling Lan, Chong Luo, *Senior Member, IEEE*, Enhong Chen, *Senior Member, IEEE*, Feng Wu, *Fellow, IEEE*, and Wenjun Zeng, *Fellow, IEEE*

***Abstract*—We propose a progressive pseudo-analog video transmission scheme that simultaneously handles SNR and bandwidth variations with graceful quality degradation for mobile video streaming. With the inherited SNR-adaptability from pseudo-analog transmission, the proposed progressive solution acquires bandwidth adaptability through an innovative scheduling algorithm with optimal power allocation. The basic idea is to aggressively transmit or retransmit important coefficients so that distortion is minimized at the receiver after each received packet. We derive the closed-form expression of reduced distortion for each packet under given transmission power and known channel conditions, and show that the optimal solution can be obtained with a water-filling algorithm. We also illustrate through analyses and simulations that a near-optimal solution can be found through approximation when only statistical channel information is available. Simulations show that our solution approaches the performance upper bound of pseudo-analog transmission in an additive white Gaussian noise channel and significantly outperforms existing pseudo-analog solutions in a fast Rayleigh fading channel. Trace-driven emulations are also carried out to demonstrate the advantage of the proposed solution over the state-of-the-art digital and pseudo-analog solutions under a real dramatically varying wireless environment.**

***Index Terms*—Multimedia communication, radio communication, streaming media.**

I. Introduction

ACCORDING to the Cisco Visual Networking Index (VNI) [1], mobile data traffic will increase 10-fold between 2014 and 2019, with video constituting a large portion of it. In this paper, we address one of the most important mobile video applications known as mobile video streaming. The two biggest challenges facing a mobile streaming application are the dramatically varying channel conditions and the stringent latency requirements. When a mobile device is downloading and playing a video, each video frame has a playback deadline. Failing to transmit a decodable bit stream of a frame before its playback deadline not only creates unpleasant user experience, but also wastes precious network bandwidth resources. When a mobile device is recording and uploading a video, the transmission delay should also remain small to avoid local buffer overflow. Under such a stringent latency requirement, it is very challenging to provide a high quality of experience (QoE) under the time-varying wireless channel.

Contemporary digital video transmission systems adopt separate source coding and channel coding. Video source is first compressed into a bit stream through a standard video encoder, such as the most widely used H.264/AVC [2]. Then, the bit stream is encoded by a channel encoder before transmission. At the physical layer, the channel coded bits are mapped into discrete complex symbols, which are actually transmitted, in a process called modulation. This separate source channel coding framework works well when the channel condition is stable, but when the channel condition varies dramatically, it suffers from the well-known *threshold effect* and the *leveling off effect*. As explained in [3], [4], when the channel condition gets worse than a certain threshold, the receiver cannot correctly demodulate the received complex symbols and the bit error rate goes beyond the error correction capability of channel decoding. When this happens, the entire video stream may be corrupted. This is called the threshold effect. The leveling-off effect refers to the fact that no performance gain can be achieved when the actual channel condition gets better than expected. This is because, once the source rate and the transmission rate are fixed, the best achievable performance is determined.

In order to adapt to channel conditions, conventional digital solutions connect the source encoder and the transmission modules with a local buffer. The source encoder (e.g. H.264/AVC [2]) adjusts the source coding rate according to the buffer fullness. The transmission module, on the other hand, fetches data from the buffer, performs channel encoding, packetization, and then chooses the most appropriate modulation rate to send the packets out. This design has three drawbacks. First, the basic method of a digital video encoder adjusting the source rate adjusts the quantization parameter, which is very coarse and sometimes unpredictable. Second, the efficiency of adaptive modulation and coding (AMC) relies heavily on the timeliness and precision of channel feedback, which is hard

Manuscript received January 27, 2016; revised July 18, 2016 and February 20, 2017; accepted March 17, 2017. Date of publication March 23, 2017; date of current version July 15, 2017. This work was supported in part by the Natural Science Foundation of China under Contract 61631017. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Christian Timmerer. *(Corresponding author: Chong Luo.)*

D. He was with Microsoft Research Asia, Beijing 10080, China. He is now with the University of Science and Technology of China, Hefei 230026, China (e-mail: hedl@mail.ustc.edu.cn).

C. Lan, C. Luo, and W. Zeng are with Microsoft Research Asia, Beijing 10080, China (e-mail: culan@microsoft.com; cluo@microsoft.com; wezeng@microsoft.com).

E. Chen and F. Wu are with the University of Science and Technology of China, Hefei 230026, China (e-mail: cheneh@ustc.edu.cn; fengwu@ustc.edu.cn).

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

Digital Object Identifier 10.1109/TMM.2017.2686703

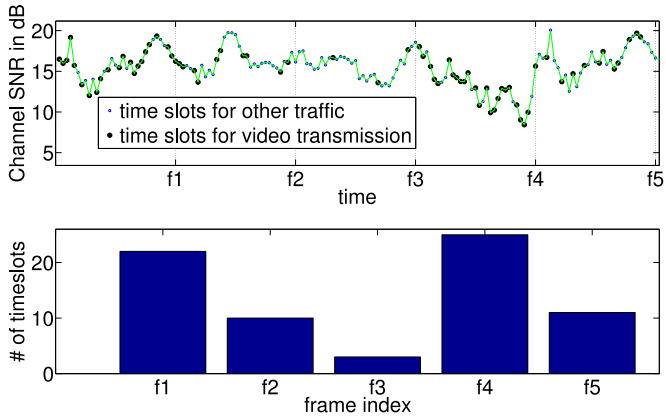


Fig. 1. Illustrating the challenges facing a mobile video streaming application: varying channel SNR and varying bandwidth share.

to obtain. Third, using a local buffer to connect the source and the channel creates a difficult engineering tradeoff between the channel adaptation capability and the amount of induced delay.

Recently, a pseudo-analog video transmission scheme named SoftCast [5], [6] has attracted much attention. It skips quantization and entropy coding in the conventional digital video encoder, and directly transmits the power-scaled DCT coefficients through amplitude modulation (amplitude shift keying, to be precise). Although pseudo-analog transmission does not explicitly perform compression, it achieves a similar effect through power allocation. Specifically, the result of traditional compression, which adopts quantization and entropy coding, redistributes the bits in a data stream. DCT coefficients with larger variances are allocated with more bits, while those with smaller variances are allocated with fewer bits. All bits are treated almost equally during transmission in terms of occupied time slots and transmission power. Thus, the redistribution of bits among DCT coefficients translates to a redistribution of bandwidth and power. SoftCast achieves the analogous effect by a direct redistribution of power among the DCT coefficients. In addition, when bandwidth is not sufficient, it will discard the least important coefficients (i.e. those with the smallest variances) to achieve bandwidth compaction. In SoftCast transmission, when channel noise perturbs the transmitted signal samples, the perturbation naturally translates into approximation in the original video pixels. As a result, a receiver can reconstruct the video at a quality that is commensurate with its channel quality, and the degradation is graceful with decreased channel SNR.

However, the time-varying characteristic of wireless channels is reflected in both SNR and bandwidth variations. Fig. 1 shows the SNR trace we collected in a real wireless environment in around 160 ms time frame (where five video frames are collected and to be transmitted) and a possible bandwidth allocation. Note that the bandwidth variation is prominent in a multi-user system due to the shared nature of the wireless medium. While pseudo-analog video transmission is inherently SNR-adaptable, how does it perform in a bandwidth-varying environment?

Unfortunately, the current design of pseudo-analog video transmission cannot gracefully handle bandwidth variation, especially in a low-latency setting. Specifically, the current design requires that the available bandwidth for each group of picture (GOP) is known in advance. When the channel bandwidth is smaller than the source bandwidth, the sender discards the DCT coefficients with the least importance (or the smallest variances). Then, the remaining coefficients are power scaled according to their variances and the total power budget. Since the coefficients with different variances need dramatically different transmission power, sometimes by several degrees of magnitude, they need to be mixed in order to form PHY packets of the same power. As a result, each packet contains both important and not-so-important coefficients. During transmission, if the available bandwidth is smaller than expected, some of the packets may not have the opportunity to transmit. When this happens, the important coefficients in these packets are discarded and the received video quality will be dramatically degraded.

In this paper, we propose a progressive pseudo-analog video transmission scheme, with the objective of simultaneously handling SNR and bandwidth variations in a timely and graceful manner. The basic idea is as simple as transmiting the DCT coefficients successively according to their importance. However, the importance of a coefficient is determined by its variance, which is exactly the transmission power in amplitude modulation. Grouping important coefficients together will create extremely high-power packets which is impossible for a practical system to transmit. To address this challenge, we suppress the transmission power of important coefficients in any single transmission, but allow for re-transmission of these coefficients so that the power from multiple transmissions can be accumulated. At the receiver, video reconstruction can be performed at any moment, and preferably right before the playback deadline of a GOP. This is in contrast to the digital transmission in which the receiver has to wait for all the packets belonging to a specific GOP before it starts decoding. With this feature, the proposed progressive transmission scheme can operate in a "rebuffer-free" mode with a much smaller start-up delay than conventional digital schemes. Then the problem is, whenever there is a transmission opportunity, determining the coefficients (or chunks) to be included in the packet and their transmission powers so that the overall distortion is minimized at the receiver when this packet is received. A scheduling algorithm and a power allocation algorithm need to be designed to solve this problem.

In our preliminary work [7], we have verified this retransmission idea in an additive white Gaussian noise (AWGN) channel. In this paper, we extend our design to handle fading channels and propose a fine-grained scheduling algorithm. We implement the proposed progressive solution and evaluate the results through extensive simulation and emulation. Simulations validate that, in the AWGN channel, our progressive transmission scheme outperforms SoftCast by a notable margin. It closely approaches the performance upper bound of a pseudo-analog scheme in which the actual bandwidth is assumed to be known in advance. Simulations also show that, the proposed scheme achieves the design goal in fast fading channels, allowing a receiver to recover the video at a quality that is commensurate with its instantaneous SNR and bandwidth share. Trace-driven evaluations are also performed based on a software defined radio platform SORA [8]. Results show that in a realistic wireless environment, the proposed scheme outperforms SoftCast and a state-of-the-art digital solution (scalable

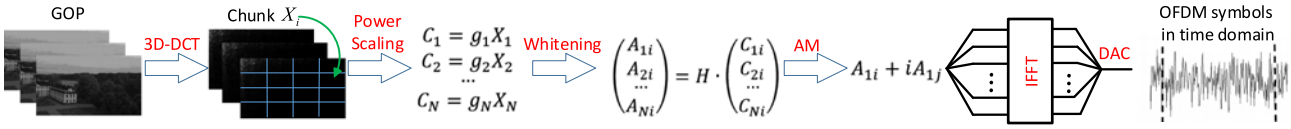


Fig. 2. Sender-side flowchart of a pseudo-analog video transmission system named SoftCast [5]. IFFT stands for "inverse fast Fourier transform," which is used to perform the orthogonal frequency division multiplexing (OFDM) process and DAC stands for "digital to analog converter."

video coding + 802.11PHY) in terms of both average and worst-case performance.

The rest of this paper is organized as follows. Section II reviews related work, including the digital and pseudo-analog solutions for mobile video streaming. The overall framework and problem formulation are described in Section III. Section IV presents our proposed solution. The evaluation results, in both simulated and real wireless environments, are presented in Section V. We conclude the paper in Section VI.

II. Related Work

A. Digital Video Coding and Transmission

Most conventional digital video communication systems adopt a separate source-channel design for implementation convenience and optimality. Specifically, source coding is standardized to follow motion estimation, discrete cosine transform, quantization and entropy coding processes. Currently, H.264 [2] is the most widely used video coding standard although the newest HEVC standard [9] achieves more bit savings. The transmission module treats all bits equally and does its best to ensure error-free transmission. To do so in a time-varying wireless channel, channel coding is adopted and an appropriate modulation scheme is selected.

Both the source and the channel provide adaptive and progressive solutions. The source encoder can adjust the source coding rate by varying the quantization parameter QP. A larger QP results in a lower rate bit stream, and vice versa. The scalable video coding (SVC) extension [10] of H.264/AVC further provides a progressive solution. SVC encodes the source video into the base layer and several enhancement layers. The base layer alone is decodable, but more successive enhancement layers will improve the decoded video quality. In the channel, the adaptive modulation and coding (AMC) mechanism is standardized in major physical layer protocols, including IEEE 802.11 [11] and LTE [12]. Furthermore, the combination of hybrid automatic request retransmission (HARQ) [13] and rate-compatible channel codes [14] could provide a better channel adaptation capability through progressive transmission.

In a video streaming application, the source and channel should be connected by a buffer and a buffer management mechanism is needed. In some previous work [15], [16], the source encoder adopts adaptive coding, which adjusts the quantization parameter QP in order to avoid the buffer from overflow or underflow. In another work [17], the source adopts SVC and the buffer manager is responsible for discarding certain enhancement layers when the effective transmission rate is low.

However, these works suffer from some or all of the following drawbacks. First, in SVC, there is a tradeoff between source coding efficiency and granularity of rate adaptation. In adaptive source coding, there are many ways to adapt to the bandwidth requirement at the encoder. One commonly used approach is to adjust the quantization parameter, which is coarse and sometimes unpredictable. Second, the effective rate of digital channel coding and modulation relies heavily on the timeliness and precision of channel feedback, which is hard to acquire. When the actual channel condition is better than expected, no additional gain can be obtained; when the actual channel condition is worse, the received wireless symbol is almost useless. The digital progressive transmission only provides a remedy, not a cure. Third, if the adaptive source coding approach is adopted (not SVC), determining the size of the local buffer is very important to the system performance, but is quite tricky as well. A long buffer offers better adaptation capability but induces longer delay, and a short one offers the opposite.

B. Pseudo-analog Video Processing and Transmission

Recently, a new pseudo-analog video transmission paradigm has emerged. The pioneering work is SoftCast [5] and its sender-side flowchart is shown in Fig. 2. For each group of pictures (GOP), three dimensional discrete cosine transform (3D-DCT) is performed. The transform coefficients are partitioned into equal-sized chunks (denoted by X_i in Fig. 2) and the variance for each chunk is computed. The coefficients in each chunk are treated as instances drawn from the same zero-mean Gaussian distribution. Normally, chunks belonging to the low frequency bands have much higher energy (variance) than those belonging to the high frequency bands. If the available bandwidth is not sufficient for transmitting all the coefficients, chunks with smaller variances will be discarded.

Then, the power scaling factors ($g_1, g_2 ... g_N$ in Fig. 2) for the chunks to be transmitted are computed. According to Jakubczak and Katabi [5], the optimal power scaling factor for each chunk that minimizes the overall distortion should be inversely proportional to the square root of the chunk's standard deviation. Nevertheless, the scaled coefficients (denoted by $C_1, C_2 ... C_N$ in Fig. 2) may still differ by several orders of magnitude. Therefore, they are whitened by the Hadamard transform before being transmitted with amplitude modulation (AM)[1] over the raw OFDM channel. At the receiver, a minimum mean square error (MMSE) decoder is adopted to estimate the coefficients before inverse 3D-DCT is applied to get the reconstructed frames.

The pseudo-analog video transmission framework is essentially a joint source-channel design. The basic idea is to skip

[1]The AM can be implemented by the amplitude shift keying (ASK) using the digital transceivers, and therefore the scheme is called a pseudo-analog scheme in some related work.

quantization and entropy coding in the source encoder, and let the power scaling operation provide appropriate channel protection to coefficients of varying importance. As such, the transmitted signal becomes linearly related to the pixels luminance. When noise perturbs the transmitted signal samples, the perturbation translates into approximation in the original video pixels [5]. This new video transmission framework allows a receiver to reconstruct the video with a quality that is commensurate with its channel noise level. Extensive simulations and emulations have shown that a fine-grained quality adaptation can be achieved with varying channel SNR.

In virtue of the strong channel adaptation capability, SoftCast has attracted a lot of attention, and there are many follow-up works that focus on improving the reconstructed video quality. Xiong *et al.* [18] propose an adaptive chunk division method to provide a better modeling of the coefficients. Cui *et al.* [19] modify the framework such that denoising techniques can be applied to improving the reconstructed video quality. Xiong *et al.* [20] and He *et al.* [4] propose different signal processing and power allocation schemes to improve the received subjective visual quality. Liu *et al.* propose ParCast for video unicast in a MIMO-OFDM system [21], [22]. HDAVT [23] design the optimal resource allocation in slow fading channel. However, these works either optimize performance for AWGN channels or assume that the precise channel fading parameters can be obtained, and none of them consider the bandwidth adaptability.

There do exist some efforts to address all kinds of heterogeneity in the pseudo-analog transmission framework. In [3], [24], video is divided into a base layer and enhancement layers to provide resolution scalability. Fan *et al.* [25] address bandwidth heterogeneity by means of layered coset coding. However, it is designed based on an important assumption that the channel bandwidth is no less than the source bandwidth. Again, these works target at a predictable wireless environment. The only work in literature that potentially handles both fast fading channel and bandwidth variation is [26], although they do not explicitly take bandwidth adaptation into consideration. The basic idea is to retransmit some important chunks to obtain channel diversity gain and combat fading. While the basic idea of our proposed scheme is also retransmission, our scheme is designed and optimized to handle large bandwidth variations, which is commonly seen in mobile video streaming applications.

III. Progressive Pseudo-analog Video Transmission

A. Framework Overview

The progressive pseudo-analog video transmission is designed to be both SNR-adaptable and bandwidth-adaptable. While SNR-adaptability is an inborn trait of pseudo-analog transmission, bandwidth-adaptability is achieved through the proposed greedy scheduling algorithm that allows retransmission of important chunks. Fig. 3 illustrates the proposed framework at the sender. It is a cross-layer design involving the application layer (APP), the medium access control (MAC) and the physical layer (PHY). Our unique designs are highlighted in a bold dashed box in Fig. 3.

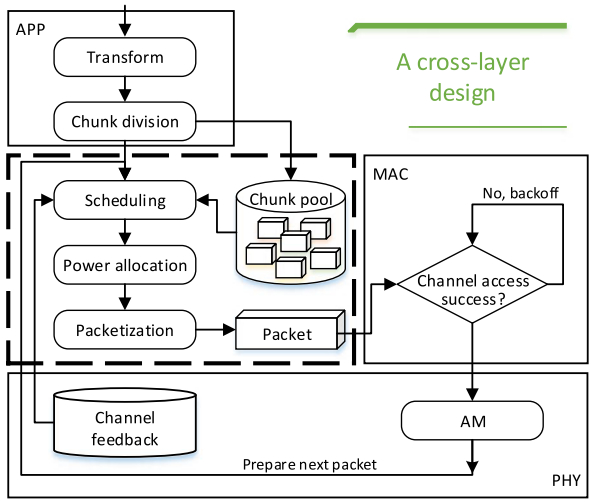


Fig. 3. Progressive pseudo-analog video transmission framework.

In APP, a decorrelation transform (e.g. 3D-DCT) is performed over the input video frames and the transform coefficients are divided into chunks. Coefficients in one chunk are treated as instances drawn from the same zero-mean Gaussian distribution. For simplicity and without loss of generality, we consider equal-sized rectangular-shape chunk division. All chunks are put into a chunk pool and the variance of each chunk is computed. Generally speaking, the variance of a chunk reflects its importance.

The scheduling module is the key module in our design. It picks K chunks out of the total N chunks to compose a *tile*. The size of K is determined such that each tile can fit into a single PHY packet. The power allocation module serves two purposes. First, it ensures that the average symbol power of the packet does not exceed the given power budget. Second, it allocates power among different chunks to minimize distortion. Both scheduling and power allocation decisions are made with an objective to minimize the total mean squared error (MSE) at the receiver after this packet is received, and by taking the previous channel feedback into consideration. At the MAC, the sender waits to access the wireless medium. Once a packet is successfully sent out, the scheduling model will prepare for the next packet. If the playback deadline for the current GOP is about to pass (in the case of downlink streaming) or the next GOP has been recorded in the buffer (in the case of uplink uploading), the sender will move to process the next GOP.

Note that, in the proposed framework, an important chunk can be transmitted multiple times. This never happens in the original SoftCast system. This is because SoftCast is optimized for an AWGN channel only. In an AWGN channel model, transmitting a signal for n times using power $P_1, P_2, \ldots P_n$ results in the same MSE as transmitting it once using power $\sum_{i=1}^{n} P_i$. Although the power allotments for different chunks are dramatically different, SoftCast mixes instances from different chunks into one packet to ensure that the average power of each packet satisfies a given constraint. However, if a sender is not sure whether it has a chance to transmit one more entire packet for a given GOP, it should aggressively transmit the important chunks, under the

power budget of each packet. If later there are more transmission opportunities, these important chunks should be re-transmitted so that the power can be accumulated to approach optimal power allocation. If there are even more transmission opportunities, some less important chunks can also be transmitted. This is the basic idea behind our progressive design.

The receiver collects as many copies of each coefficient as possible for a GOP before its playback deadline or the sender stops transmitting it. Then, multiple copies of the same coefficient are merged by maximum ratio combining (MRC) before minimum mean squared error (MMSE) decoding is performed. Finally, the inverse transform reconstructs the video frames.

B. System Model and Problem Formulation

Let us consider a video sequence with frame resolution $W \times H$, where W and H indicate width and height, respectively. Assume that a GOP has F frames (F can be as small as 1 when the application does not tolerate any delay). After the decorrelation transform, the transform coefficients of each frame are partitioned into M chunks, so that a GOP has $N = M \times F$ chunks in total. The size of each chunk is denoted by L, which is equal to $(W \times H)/M$. We sort these chunks in descending order according to their variances and re-index them from 1 to N. The source chunks are denoted by S_i, $i = 1...N$. Usually, chunks located in the lower frequency band have larger variances. Let λ_i denote the variance of the ith chunk and we have $\lambda_{i_1} \geq \lambda_{i_2}$, $\forall i_1 \leq i_2$.

Let K be the number of chunks which fit into one PHY packet, and let Γ be the maximum number of packets that can be transmitted for a particular GOP. Without loss of generality, we consider a scenario where $\Gamma = N/K$. We use $S_{c,i}$ to denote the complex source chunk and it is formed by combining the odd part $S_{i,o}$ and even part $S_{i,e}$ of S_i

$$S_{c,i} = \frac{S_{i,o} + jS_{i,e}}{\sqrt{2}}, i = 1, 2, ...N. \tag{1}$$

Suppose after the scheduling process, the ith tile T_i is composed of K chunks whose indexes are $\Omega(i) = [i_1, i_2, ..., i_K]$. The corresponding power scaling factors of T_i are denoted by $G_i = diag\{g_{i_1,R_{i_1}}, g_{i_2,R_{i_2}}, ..., g_{i_K,R_{i_K}}\}$, where R_{i_k} denotes the total number of times that the i_kth chunk has been transmitted. The whitening matrix, denoted by W_i, is a $K \times K$ unitary Hadamard matrix. Since these K chunks are transmitted in the same packet, we could simply assume that they have the identical channel gain. Let h_i denote the complex channel parameter that tile T_i experiences. Given the above notations, when there are $\Gamma_0 \leq \Gamma$ tiles transmitted, we can write the received signals Y in the following matrix form:

$$Y = HMGQX + V \triangleq AX + V \tag{2}$$

where $X = [S_{c,1}, S_{c,2}..., S_{c,N}]^T$ denotes the source signal and V is the additive Gaussian channel noise with power σ_n^2. $M = blkdiag\{W_1, W_2, ..., W_{\Gamma_0}\}$ is a block diagonal matrix which performs tile-wise whitening, and $G = blkdiag\{G_1, G_2, ..., G_{\Gamma_0}\}$ is the diagonal power scaling matrix. $H = blkdiag\{H_1, H_2, ..., H_{\Gamma_0}\}$ denotes the channel parameters, where $H_i = h_i I_{K\times K}$ and $I_{K\times K}$ is the identical matrix. Q is the scheduling matrix composed of zeros and ones. For example, if $K = 2, N = 8$ and the scheduling results are $\Omega(1) = [1, 2], \Omega(2) = [1, 3], \Omega(3) = [2, 4]$, then

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

and $G = diag\{g_{1,1}, g_{2,1}, g_{1,2}, g_{3,1}, g_{2,2}, g_{4,1}\}$.

The MMSE decoder C reconstructs the source with minimum mean square error

$$\hat{X} = CY = \Lambda A^H (A\Lambda A^H + \sigma_n^2 I)^{-1} Y \tag{3}$$

where $\Lambda = diag\{\lambda_1, \lambda_2, ..., \lambda_N\}$ and $(\cdot)^H$ denotes Hermitian transpose.

In our progressive transmission framework, the scheduling algorithm picks K chunks at a time to form a tile (packet). For any integer i in $\{1, 2, ..., \Gamma\}$, in the ith round, the scheduling task is to find $\Omega(i)$ and G_i to minimize the expected MSE after T_i is received, given the entire sender-side information and available receiver-side information of the previous $i-1$ tiles. Mathematically, the scheduling objective in the ith round can be expressed as

$$\min_{G,Q} \; E\{(\widehat{X_i} - X)^T (\widehat{X_i} - X)\}$$
$$\text{s.t.} \sum_{k=1}^{K} g_{i_k,R_{i_k}}^2 \lambda_{i_k} \times L \leq K \times E_s \times L \tag{4}$$

where $\widehat{X_i}$ is the reconstruction of X after tile T_i is received. Notation $E\{\cdot\}$ denotes the expectation. The constraint in (4) means that each tile (packet) is energy constrained, and E_s is the average symbol energy.

IV. The Proposed Solution

In the proposed progressive framework, chunk scheduling and power allocation are two coupled problems. In this section, we first derive the optimal power allocation within a tile when the chunks have been determined, and then present the scheduling algorithm which decides what chunks should be included in a tile. In addition, when only partial receiver-side information is known, we discuss the necessary approximations in implementing our algorithm.

A. Power Allocation

Each PHY packet contains one and only one tile, so all tiles have the same power budget. For each of the K chunks scheduled to tile T_i, the power scaling factors $g_{i_k,R_{i_k}}$ should be determined in order to minimize the expected total distortion of all the N chunks at the receiver. Note that some of the chunks in tile T_i may have been transmitted in previous packets. Therefore, the quality of previous transmissions, if available, should be taken into account.

First, we derive the distortion of transmitting source $S_{c,i}$ with variance λ_i R_i times when the additive channel noise power is σ_n^2. Suppose the channel parameters it experiences are $\bar{H}_i = diag\{h_{i,1}, h_{i,2}, ..., h_{i,R_i}\}$, the received signals are R_i noisy versions of $S_{c,i}$

$$Y_i = \bar{H}_i \bar{G}_i S_{c,i} + V_i \tag{5}$$

where $V_i = [v_{i,1}, v_{i,2}, ..., v_{i,R_i}]^T$ denotes additive noise and $\bar{G}_i = [g_{i,1}, g_{i,2}, ..., g_{i,R_i}]^T$ denotes power scaling factors. To leverage channel diversity and improve channel quality, maximum ratio combining (MRC) is applied before decoding the source, thus

$$\tilde{S_{c,i}} = \frac{(\bar{H}_i \bar{G}_i)^H}{\|\bar{H}_i \bar{G}_i\|^2} Y_i = S_{i_k} + \tilde{V}_i \tag{6}$$

where $\tilde{V}_i = \frac{(\bar{H}_i \bar{G}_i)^H}{\|\bar{H}_i \bar{G}_i\|^2} V$ and its variance equals $\sigma_n^2 / \|\bar{H}_i \bar{G}_i\|^2$. $(\cdot)^H$ denotes the Hermitian transpose. MMSE estimation is used to detect the transmitted signal, as has been derived in [26], the distortion (or MSE) of transmitting $S_{c,i}$ R_i times becomes

$$\zeta(S_{c,i}, R_i) = \frac{\lambda_i \sigma_n^2}{\sum_{l=1}^{R_i} \|h_{i,l}\|^2 g_{i,l}^2 \lambda_i + \sigma_n^2}. \tag{7}$$

Now we consider the case of transmitting a tile T_i which consists of chunks $S_{c,i_1}, S_{c,i_2}, ..., S_{c,i_K}$. Assume that chunk S_{c,i_k} has already been transmitted R_{i_k} times in the previous transmission opportunities, and its previous transmission state information (*PTSI*), i.e., $g_{i_k,1}, g_{i_k,2}, ..., g_{i_k,R_{i_k}}$ and $h_{i_k,1}, h_{i_k,2}, ..., h_{i_k,R_{i_k}}$ are available, according to (7), the distortion of tile T_i can then be derived as

$$\begin{aligned} \zeta(T_i) &= \sum_{k=1}^{K} \zeta(S_{c,i_k}, R_{i_k} + 1) \\ &= \sum_{k=1}^{K} \frac{\lambda_{i_k} \sigma_n^2}{\left(\sum_{l=1}^{R_{i_k}} \|h_{i_k,l}\|^2 g_{i_k,l}^2 \lambda_{i_k}\right) + \sigma_n^2 + \|h\|^2 g_{i_k}^2 \lambda_{i_k}} \end{aligned} \tag{8}$$

where h is the channel gain that the transmitted tile is to experience and g_{i_k} is the power scaling factor of chunk S_{c,i_k} which is to be optimized.

Note that our goal is to minimize the total distortion of all N chunks such that at the transmission of tile T_i, the receiver can gain optimal performance. When the *PTSI* is given and the K chunks are fixed for composing tile T_i, to minimize the distortion of K chunks is equivalent to minimizing the total distortion. Therefore, to solve the optimal power allocation problem when K chunks and their *PTSI* are given, the objective can be formulated as follows:

$$\begin{aligned} &\min_{g_{i_k}} \sum_{k=1}^{K} \frac{\lambda_{i_k} \sigma_n^2}{\left(\sum_{l=1}^{R_{i_k}} \|h_{i_k,l}\|^2 g_{i_k,l}^2 \lambda_{i_k}\right) + \sigma_n^2 + \|h\|^2 g_{i_k}^2 \lambda_{i_k}} \\ &\text{s.t.} \sum_{k=1}^{K} g_{i_k}^2 \lambda_{i_k} \le K * E_s. \end{aligned} \tag{9}$$

The solution of this optimization problem can be obtained by the method of Lagrange multiplier and the water-filling algorithm. The closed-form expression can be derived as

$$\begin{cases} g_{i_k} = \left(\sqrt{\frac{\sigma_n}{\sqrt{\nu}\|h\|^2 \lambda_{i_k}}} - \frac{A_{i_k}}{\|h\|^2 \lambda_{i_k}}\right)^+ \\ \text{s.t.} \sum_{k=1}^{K} g_{i_k}^2 \lambda_{i_k} \le K * E_s \end{cases} \tag{10}$$

where ν is some constant and is chosen to meet the total energy constraint as shown in (10), A_{i_k} equals $\sum_{l=1}^{R_{i_k}} \|h_{i_k,l}\|^2 g_{i_k,l}^2 \lambda_{i_k} + \sigma_n^2$, and the operator $(a)^+$ is defined as $\max\{0, a\}$.

In this subsection, in order to derive the power allocation strategy, we have assumed the availability of channel state information (CSI), including the channel gain $\|h\|^2$ and the noise power σ_n^2, of each previous packet. Both parameters can be estimated at the receiving station and be feedback to the sending station through the reverse channel. In practice, both parameters can be attached to the end of the acknowledgement (ACK) frame which is sent from the receiving station to the sending station after each data frame is received. For example, if each channel parameter is represented by 16 bits, the overhead on the reverse channel is around 15 kbps, which is quite small. This overhead can be further reduced if the CSI is reported once every a few packets or the background noise power is approximated by the noise of the reverse channel.

B. Scheduling

In the proposed progressive transmission framework, the transform coefficients of a GOP are divided into equal-sized chunks which are put into a chunk pool. The scheduling task is to pick K chunks from the pool to form a tile (packet) for the next transmission opportunity. We consider a highly dynamic environment where the sender does not know how many more packets can be transmitted for the current GOP. Therefore, the scheduling algorithm adopts a greedy approach. Specifically, it tries to minimize the receiver-side distortion after the scheduled packet is received. In other words, the scheduling problem can be defined as determining what K chunks should be included in the current tile such that the largest distortion reduction can be achieved given the past and current channel conditions.

A straightforward way to solve the scheduling problem is to exhaustively search all possible combinations of K chunks and compare their expected distortions under the corresponding optimal power allocation. However, the total number of possible combinations is C_N^K. It is inefficient to perform power allocation for each of the combinations, so the challenge is how to reduce the computational complexity of the scheduling algorithm without breaching its optimality. To address this challenge, we first present a proposition.

Proposition 1: Let $S^{(1)}, S^{(2)}, ..., S^{(M)}, K \le M \le N$ be the collection of chunks which have never been transmitted in the chunk pool and their variances satisfy $\lambda^{(1)} \ge \lambda^{(2)} \ge, ..., \lambda^{(M)}$. If K chunks are to be chosen from these M chunks to form a tile, the optimal selection that minimizes the overall distortion of all N chunks is to choose $S^{(1)}, S^{(2)}, ..., S^{(K)}$.

Proof: To prove that selecting $S^{(1)}, S^{(2)}, ..., S^{(K)}$ is optimal, we only have to prove that substituting any one of these K chunks by a chunk $S^{(j)}, j > K$, the total distortion of $S^{(1)}, S^{(2)}, ..., S^{(M)}$ will not decrease. Without loss of generality, we can assume $S^{(K)}$ is replaced by $S^{(j)}$, and $g^{(1)}, g^{(2)}, ..., g^{(K-1)}, g^{(j)}$ are the optimal power allocation factors for transmitting $S^{(1)}, S^{(2)}, ..., S^{(K-1)}, S^{(j)}$, note that the power scaling factors should satisfy that $\sum_{k=1}^{K-1}(g^{(k)})^2\lambda^{(k)} + (g^{(j)})^2\lambda^{(j)} \leq KE_s$, which is the total power constraint.

Then, the distortion of all the M chunks if we transmit $S^{(1)}, S^{(2)}, ..., S^{(K-1)}, S^{(j)}$ becomes

$$D_0 = \sum_{k=1}^{K-1} \frac{\lambda^{(k)}\sigma_n^2}{\|h\|^2(g^{(k)})^2\lambda^{(k)} + \sigma_n^2} + \frac{\lambda^{(j)}\sigma_n^2}{\|h\|^2(g^{(j)})^2\lambda^{(j)} + \sigma_n^2} + \sum_{m=K, m\neq j}^{M} \lambda^{(m)} \tag{11}$$

where $\|h\|^2$ is the channel gain of this tile. The first two terms in (11) denote the distortion of the chunks which will be transmitted and the last term denotes the distortion of the chunks which are not to be transmitted.

Now, we consider transmitting $S^{(1)}, S^{(2)}, ..., S^{(K)}$. We use the power allocation factors $g^{(1)}, g^{(2)}, ..., g^{(K-1)}, \widehat{g^{(j)}}$, where $\widehat{g^{(j)}} = \frac{g^{(j)}\sqrt{\lambda^{(j)}}}{\sqrt{\lambda^{(K)}}}$. In this case, the total power constraint $\sum_{k=1}^{K-1}(g^{(k)})^2\lambda^{(k)} + (\widehat{g^{(j)}})^2\lambda^{(K)} \leq KE_s$ is met. Please keep in mind that these power allocation factors are not necessarily optimal for transmitting $S^{(1)}, S^{(2)}, ..., S^{(K)}$. Using D_1 to denote the distortion of all the M chunks if we optimally allocate power to these K chunks, and $\widehat{D_1}$ to denote the total distortion if we use $g^{(1)}, g^{(2)}, ..., g^{(K-1)}, \widehat{g^{(j)}}$ as their corresponding power scaling factors, by definition we have

$$D_1 \leq \widehat{D_1} \tag{12}$$

$$\widehat{D_1} = \sum_{k=1}^{K-1} \frac{\lambda^{(k)}\sigma_n^2}{\|h\|^2(g^{(k)})^2\lambda^{(k)} + \sigma_n^2} + \frac{\lambda^{(K)}\sigma_n^2}{\|h\|^2\left(\widehat{g^{(j)}}\right)^2\lambda^{(K)} + \sigma_n^2} + \sum_{m=K+1}^{M} \lambda^{(m)}. \tag{13}$$

Combining (11) and (13), the difference of D_0 and $\widehat{D_1}$ can be calculated

$$\widehat{D_1} - D_0 = \frac{\lambda^{(K)} - \lambda^{(j)}}{1 + \frac{\|h\|^2\left(\widehat{g^{(j)}}\right)^2\lambda^{(j)}}{\sigma_n^2}} + \lambda^{(j)} - \lambda^{(K)}. \tag{14}$$

Because $\lambda^{(K)} \geq \lambda^{(j)} \geq 0$, we can conclude that $\widehat{D_1} - D_0 \leq 0$. Recall that in (12) we get $D_1 \leq D_0$, meaning Proposition 1 is true. ■

Proposition 1 indicates that we do not need to search all N chunks to compose a tile. If chunks $S_1, ...S_U$ have been transmitted at least once in the previous transmissions, we only need to consider chunks $S_1, ..., S_{U+K}$ in the current transmission opportunity and pick K chunks from them.

Algorithm 1: The proposed scheduling algorithm for progressive transmission

Data: $\Lambda = \{\lambda_i\}, i = 1, 2, \cdots, N$; E_s, σ_n^2; K;
Result: Tile compositions: $\Omega(1)$, $\Omega(2)$, $\cdots$, $\Omega(\Gamma)$;
Power allocations: $\mathcal{G}_R$;
Initialization: $U = 0$; $R = [0, 0, ..., 0]$; $\mathcal{G}_R = \Phi$;
$\mathcal{U} = \Phi$; //the set of non-selected chunk ids
for i = *1 to* Γ **do**
  //initiate the K chunk ids
  $\Omega(i) = [U+1, U+2, ..., U+K]$;
  $\mathcal{U} = \{1, 2, ..., U\}$; //if $U = 0, \mathcal{U} = \Phi$
  $R_{tmp} = R$; $R_{tmp}[\Omega(i)] = R_{tmp}[\Omega(i)] + 1$;
  $\mathcal{G}_{R_{tmp}} = \mathcal{G}_R$;
  Calculate $g_{\Omega(i)}$ according to (10) and update $\mathcal{G}_{R_{tmp}}$;
  $D_{min} = \varepsilon_t(\mathcal{G}_{R_{tmp}}, R_{tmp})$; $\mathcal{G}_0 = \mathcal{G}_{R_{tmp}}$; $R_0 = R_{tmp}$;
  if $\mathcal{U} \neq \Phi$ **then**
    for $v = U + K; v > U; v--$ **do**
      for *each* u *in* $\mathcal{U}$ **do**
        $\widetilde{\Omega(i)} = \Omega(i)$;
        Replace v in $\widetilde{\Omega(i)}$ by u;
        $R_{tmp} = R$; $R_{tmp}[\widetilde{\Omega(i)}] = R_{tmp}[\widetilde{\Omega(i)}] + 1$;
        $\mathcal{G}_{R_{tmp}} = \mathcal{G}_R$;
        Calculate $g_{\widetilde{\Omega(i)}}$ according to (10) and update $\mathcal{G}_{R_{tmp}}$;
        $D_t = \varepsilon_t(\mathcal{G}_{R_{tmp}}, R_{tmp})$;
        if $D_t < D_{min}$ **then**
          $D_{min} = D_t$; $\Omega(i) = \widetilde{\Omega(i)}$;
          $\mathcal{G}_0 = \mathcal{G}_{R_{tmp}}$; $R_0 = R_{tmp}$;
        end
      end
      $\mathcal{U} = \{1, 2, ..., U, U+1, ..., U+K\} - \{u_0 | u_0 \ is \ in \ \Omega(i)\}$;
    end
  end
  $\mathcal{G}_R = \mathcal{G}_0$; $R = R_0$; $U = \max[\Omega(1), \Omega(2), ..., \Omega(i)]$;
end

The fast algorithm is described as follows. First, we initiate the current tile with K chunks $S_{U+1}, S_{U+2}, ..., S_{U+K}$, which have not been transmitted before. Then, we look among $S_1, S_2, ..., S_U$ for a chunk to replace S_{U+K}, if the replacement can further reduce total distortion. If such a chunk cannot be found, the process stops. Otherwise, we look for a chunk among $S_1, S_2, ..., S_U$ to replace S_{U+K-1}. We continue this process until a replacement cannot be found or all the chunks in the initial set have been replaced. In the worst case, this fast algorithm needs to evaluate UK combinations to schedule a tile. The computational complexity is greatly reduced.

Algorithm 1 presents the proposed fast scheduling algorithm. Based on the variances λ_i of the chunks, the target noise power σ_n^2, and the average transmission power E_s, the algorithm determines the tile composition for transmission opportunity i, denoted as $\Omega(i), i = 1, 2, ..., \Gamma$, and the power allocation factors. Here, $\Omega(i) = [i_1, i_2, ..., i_K]$ is the array of chunk identifiers for tile T_i. The total distortion of N chunks, denoted

by $\varepsilon_t(\mathcal{G}_R, R)$ is calculated by the following equation:

$$\varepsilon_t(\mathcal{G}_R, R) = \sum_{i=1}^{N} \frac{\lambda_i \sigma_n^2}{\left(\sum_{l=1}^{R_i} \|h_{i,l}\|^2 g_{i,l}^2 \lambda_i\right) + \sigma_n^2} \tag{15}$$

where $\mathcal{G}_R$ is the set that contains all the $g_{i,l}, i = 1, 2, ..., N;\ l = 1, 2, ..., R_i$, and $\mathbf{R} = [R_1, R_2, ..., R_N]$ records the transmission times of each chunk. $h_{i,l}$ is the channel parameter the ith chunk experiences in its lth transmission.

C. Approximation

The general progressive solution described in the previous subsection is derived under the assumption that the *PTSI*, including power scaling factors and channel parameters are all available. While the power scaling factors are determined by the sender, the channel parameters need to be measured at the receiver and the feedback usually has delay. This should not create a big problem in an AWGN channel or a slow fading channel, because $\|h\|^2$ and all $\|h_{i_k,l}\|^2$ can be treated as constants in (9) and (10). However, in a fast fading channel, $\|h\|^2$ is almost impossible to estimate, while $\|h_{i_k,l}\|^2$ can be obtained from channel feedback of the measured channel state information.

In order to allocate the power under unknown variables, the objective function of the optimal power allocation problem is changed from $\zeta(T_i)$ to the expectation of $\zeta(T_i)$, denoted by $E\{\zeta(T_i)\}$

$$\begin{aligned} &\min_{g_{i_k}} E\{\zeta(T_i)\} \\ &\text{s.t.} \sum_{k=1}^{K} g_{i_k}^2 \lambda_{i_k} \le K * E_s. \end{aligned} \tag{16}$$

In the expression of $\zeta(T_i)$ as given in (8), $\|h\|^2$ is the unknown variable. To explicitly show this, we write $\zeta(T_i)$ as $\zeta(T_i)(\|h\|^2)$. It is extremely difficult to obtain a closed-form expression of $E\{\zeta(T_i)\}$, so we propose making the following approximation:

$$E\{\zeta(T_i)(\|h\|^2)\} \approx \zeta(T_i)(E\{\|h\|^2\}). \tag{17}$$

To demonstrate that (17) is a reasonable and close approximation, we carry out some experiments for the most challenging fast Rayleigh fading channel by simulation in Matlab 2014a. We consider a case where h differs for each packet. The only information we have is the statistical distribution of the fading parameter $h \sim \mathcal{CN}(0, 1)$. Now that $\|h\|^2$ is a random variable following the Chi-squared distribution, and the probability density function (PDF) of $t = \|h\|^2$ is

$$f(t) = \frac{e^{-t/2}}{2},\ t \ge 0. \tag{18}$$

From (9), the distortion of tile T_i can be written as

$$\zeta(T_i) = \sum_{k=1}^{K} \frac{1}{a_{i_k} + b_{i_k} t} \tag{19}$$

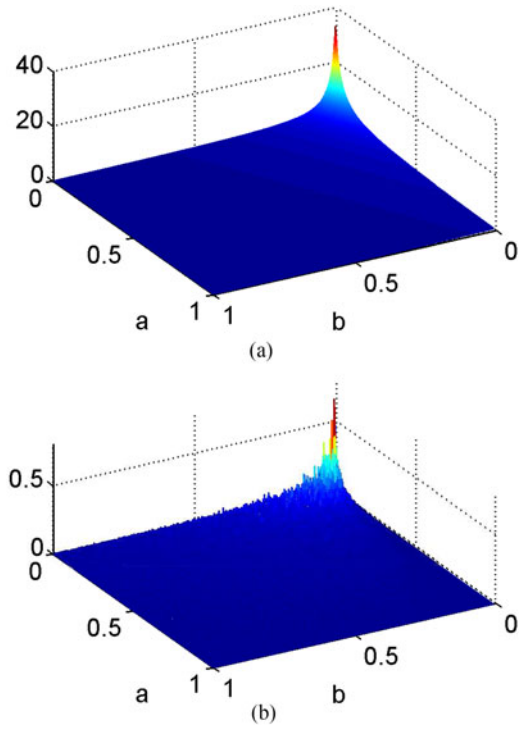


Fig. 4. Plots of Ψ and Δ. (a) $\Psi(a, b)$. (b) $\Delta(a, b)$.

where $a_{i_k} = \sum_{l=1}^{R_{i_k}} \frac{\|h_{i_k,l}\|^2 g_{i_k,l}^2}{\sigma_n^2} + \frac{1}{\lambda_{i_k}}$ and $b_{i_k} = \frac{g_{i_k}^2}{\sigma_n^2}$. Next, we define

$$\begin{aligned} \Psi(a, b) &= \int_0^{+\infty} \frac{1}{a + bt} f(t) dt \\ \widehat{\Psi}(a, b) &= \frac{1}{a + b\int_0^{+\infty} t f(t) dt} = \frac{1}{a + 2b},\ a, b > 0 \\ \Delta(a, b) &= \Psi - \widehat{\Psi}. \end{aligned} \tag{20}$$

If $\Delta(a, b)$ is small, we may conclude that (17) provides a close approximation.

We use the Monte Carlo method to calculate $\Psi(a, b)$. Specifically, we simulate the fast fading channel parameter $\|h\|^2$ according to the Chi-squared distribution in Matlab 2014a on a 64-bit Windows 10 machine. We create 5000 instances of t in total and calculate the mean value of $1/(a + bt)$. Fig. 4 plots $\Psi(a, b)$ and $\Delta(a, b)$ for $0 \le a, b \le 1$. When a and b are larger, the value of $\Psi(a, b)$ gets very small. It is obvious that $\Delta(a, b)$ is quite small in both absolute value and relative value to $\Psi(a, b)$. Therefore, if we replace $\|h\|^2$ by its mean value which is 2 in (8), it results in a close approximation of $E\{\zeta(T_i)\}$. Therefore, we can obtain the power scaling factors by substituting $\|h\|^2$ by 2 in (10). In the next section, we will show that the optimization results obtained with this approximation are very satisfactory. Till now, we have assumed the accessibility to the channel feedback of *PTSI* to obtain $\|h_{i_k,l}\|^2$. It is possible that the channel feedback delays for some time and part of $\|h_{i_k,l}\|^2$ are left unknown. We tackle this issue by replacing the unknown parameters by their mean values 2. In the next section, we will also evaluate the impact of feedback delay or even no feedback.

D. Feasibility of Real System Implementation

In fact, pseudo-analog transmission can be easily integrated into the existing 802.11 PHY layer, as described in [5]. OFDM divides the wireless spectrum into independent subcarries, some of which are pilots used for channel tracking and others are left for data transmission. Pseudo-analog transmission does not need to modify the 802.11 packet header and the pilots, therefore traditional functions of synchronization, carrier frequency offset estimation, channel estimation and phase tracking are not affected. We only need to add an option to allow the analog data to bypass FEC and QAM of PHY and to directly use the raw OFDM functionality. In addition, it is absolutely fine for the streaming media application to use the raw OFDM option while other digital communication applications continue to use the conventional standard OFDM.

The only possible issue is the input precision of the raw OFDM module. Usually, the number of input bits per symbol is limited to 14 or 16, which means that we will have to quantize the analog coefficients to 14 or 16 bits. In the next section, we will show quantitatively that the impact of this quantization is very small.

We would also like to mention that the video processing modules at both the sender and the receiver are less complex than the digital video codec H.264. Therefore, it is almost for certain that the video processing modules can run in real-time with or without hardware acceleration.

V. Evaluation

A. Settings

We implement the proposed scheme with the AWGN channel model and fast fading channel mode respectively in Matlab 2014a. In our implementation, the tile size K is set to 8 and the DCT coefficients of each frame are divided into 256 chunks. In the physical layer (PHY) of the OFDM system, the spectrum band is divided into 64 subcarriers and 48 of them are used to transmit complex analog symbols. After tile scheduling, power allocation and whitening within the tile, we pack all analog coefficients of each tile into one PLCP frame (packet), and then transmit one packet in one transmission opportunity. We also perform trace-driven testbed experiments based on the software defined radio platform SORA [8] to evaluate the progressive transmission scheme.

Test video source: In order to perform a comprehensive evaluation over videos of various contents, we use nine standard test videos as the source. The monochrome versions of these videos are used because it is well-known in the video compression community that the luminance component carries the majority of energy in a video. Besides, it will allow us to make a fair comparison with previous pseudo-analog solutions, as they all report their performances on the monochrome videos. The resolution of these video sequences is 720p (1280×720) and their frame rate is 30fps. These test videos are *stockholm, parkrun, city, spincalendar, sheriff,shuttlestart, in-to-tree, shields and jets* and they are available at Xiph test media.[2] For 720p video at a frame rate of 30fps, there are $1280 \times 720 \times 30$ real valued coefficients per second to be transmitted. In transmission systems, every (I,Q) complex symbol can transmit 2 coefficients, therefore the full source bandwidth is 13.824MHz. In our evaluation, we describe bandwidth consumption as *bandwidth ratio*, which is defined as the actual channel bandwidth over full source bandwidth.

Reference schemes: As SoftCast [5] is the state-of-the-art pseudo-analog transmission scheme in the AWGN wireless channel, in our simulation and trace-driven emulation, we choose SoftCast as one of the reference schemes. However, SoftCast is originally designed for the AWGN wireless channel. To compare performance in a fast fading channel, we choose Cui's solution [26] as a reference scheme because it is a pseudo-analog transmission scheme tailored for fast fading channels. We implement SoftCast in Matlab 2014a according to the descriptions in [5]. The implementation of Cui's solution is provided by the author. Given that both SoftCast and Cui's solution require bandwidth ratio for their power and bandwidth allocation, we implement these reference schemes by assuming a known bandwidth ratio.

As for digital solutions, considering that standard H.264 and HEVC based solutions cannot offer scalability over either channel bandwidth or channel noise level, we use a state-of-the-art digital solution based on the scalable video coding (SVC) extension of H.264 [10]. The publicly available JSVM SVC reference Software version 9.19.14[3] is used. The encoder is configured to enable inter-layer prediction.

In the implementation of the proposed scheme and all reference schemes, the GOP size is set to 8. Normally, a larger GOP size results in higher coding efficiency but longer delay. A GOP size of 8 strikes a good tradeoff between coding efficiency and low-delay requirement for video streaming applications.

Performance metric: The well-known peak-signal-noise-ratio (PSNR) objective metric is used to evaluate the reconstructed video quality and PSNR is calculated via the MSE between the received video and the groundtruth, i.e., $PSNR = 10 \log_{10} \frac{255^2}{MSE}$

B. Results in Simulated Environment

1) AWGN Channel: We first evaluate our progressive solution against SoftCast in the AWGN channel. The signal-noise-ratio (SNR) is assumed to be known. We vary the bandwidth ratio, which is defined as the ratio of channel bandwidth to source bandwidth, from 0.125 to 1, but this information is not known to the sender in both SoftCast and our solution. SoftCast simply drops packets when the bandwidth ratio is smaller than 1. We also implement an omniscient scheme based on SoftCast, which knows the exact bandwidth ratio before the transmission so that the optimal power allocation and packetization can be achieved. This omniscient scheme provides the performance upper bound for pseudo-analog video transmission.

[2]"Xiph test media." Sep. 2004, Accessed on: Jul. 2014. [Online]. Available: https://media.xiph.org/

[3]"JSVM reference software." Jun. 2011, Accessed on: Apr. 2015. [Online]. Available: http://www.hhi.fraunhofer.de/departments/video-coding-analytics/research-groups/image-video-coding/research-topics/svc-extension-of-h264avc/jsvm-refe rence-software.html

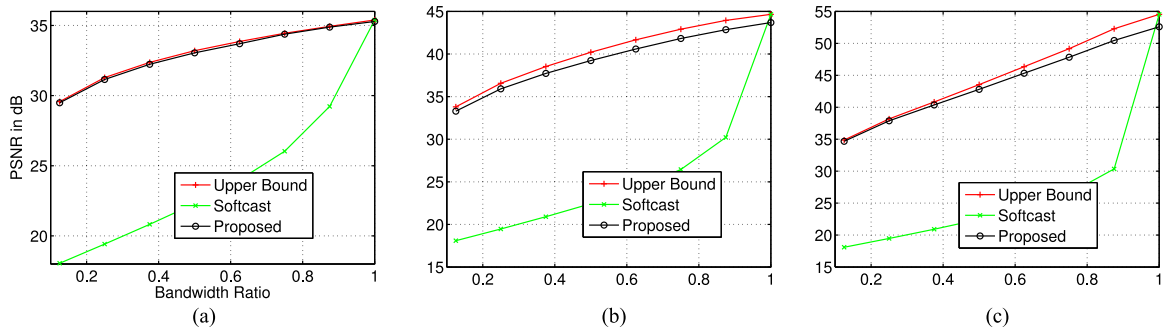


Fig. 5. Average reconstructed video PSNR of different schemes under different channel SNR and bandwidth settings. (a) $EsN0 = 5$ dB. (b) $EsN0 = 15$ dB. (c) $EsN0 = 25$ dB.

Fig. 5 shows the average received video PSNR on the nine test video sequences when the channel SNR equals to 5 dB, 15 dB and 25 dB, respectively. It is clear that SoftCast performs very poorly when the bandwidth is unknown to the sender. When the channel bandwidth is much lower than the source bandwidth, the PSNR drops dramatically. From this experiment, we may conclude that the bandwidth adaptation capability of SoftCast is very weak.

In contrast, our proposed solution achieves graceful quality degradation with the reduced bandwidth ratio, and the performance is very close to the upper bound. As shown in Fig. 5(a), when the channel SNR is 5 dB, the proposed scheme achieves almost identical performance as the upper bound. We also find from Fig. 5(b) and 5(c) that when the channel SNR is higher, say 15 dB and 25 dB, the performance gap between the proposed solution and the upper bound is very small. The gap is more noticeable when the channel condition is good and the bandwidth ratio is high. This is because our progressive solution ensures the reception quality of important chunks by sacrificing the transmission opportunity of not-so-important chunks. Nevertheless, the gap to the upper bound is only 0.96 dB when the bandwidth ratio is 1 and SNR is 15 dB.

2) Fast Rayleigh Fading Channel: Next we perform the evaluation in a fast Rayleigh fading channel. For every transmission opportunity (or packet), the fading parameter $\|h\|$ is independently and randomly generated. There are two reference schemes used for comparison. One is omniscient SoftCast, for which the bandwidth ratio is known beforehand, so that it can always perform optimal bandwidth and power allocation accordingly. The other is Cui's scheme [26], denoted as *Cui_sln*. The original design of Cui's scheme cannot adapt to bandwidth variations, so we implement two variations, assuming the bandwidth ratio to be 0.2 and 0.8 and the transmitter optimally allocate power and bandwidth accordingly. If the actual bandwidth ratio is lower than assumed, the sender randomly drops the packets; if the actual bandwidth ratio is higher, the sender retransmits randomly selected packets to make full use of the network resource.

For fairness, we do not assume channel feedback in the implementation of our solution, because both *Cui_sln* and SoftCast do not use channel state information (CSI) at the sender. We run 100 tests using the test video sequences and averaged the PSNR metric over all the test runs. Fig. 6 shows the performance of all three schemes. The shadows denote the dynamic ranges from 10 to 90 percentile performance for every scheme. It can be observed that our scheme achieves the highest average PSNR among the three schemes and the PSNR variations are quite small. Interestingly, our progressive scheme outperforms the omniscient SoftCast at all bandwidth ratio settings, and the PSNR gain is up to 1.68 dB. This is because, although the SoftCast sender knows the exact bandwidth information, the power allocation is optimized for AWGN channels. In a fast fading channel, if an important coefficient experiences deep fade, the overall performance will be greatly degraded.

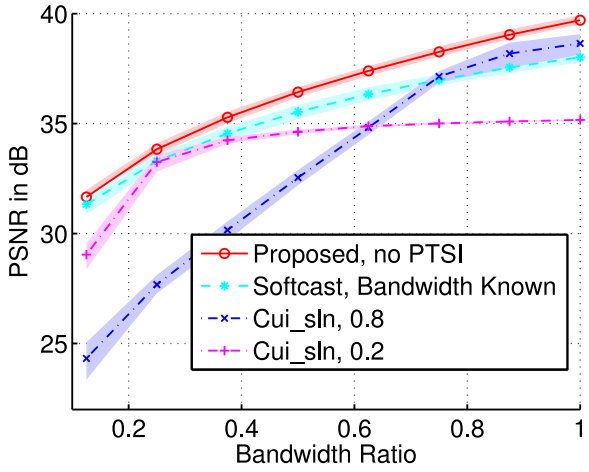


Fig. 6. Performance comparisons of difference schemes when channel SNR E_s/σ_n^2 equals 10 dB. The performance metrics are averaged over test video sequences, GOP size is 8.

Fig. 6 also shows that the bandwidth adaptation capability of Cui's solution [26] is not as good as ours. When the target bandwidth ratio is 0.2, *Cui_sln* performs well when the bandwidth ratio is relatively low, but it tends to level-off when the actual bandwidth gets larger. When the target bandwidth is 0.8, *Cui_sln* performs well at the high end, but the performance drops dramatically at the low end. This experiment clearly shows that the proposed progressive solution (even without channel feedback) outperforms state-of-the-art pseudo analog solutions in a bandwidth-varying fading channel.

C. Trace-Driven Emulation

We next evaluate the proposed scheme under real wireless environments. Evaluations are carried out using the channel

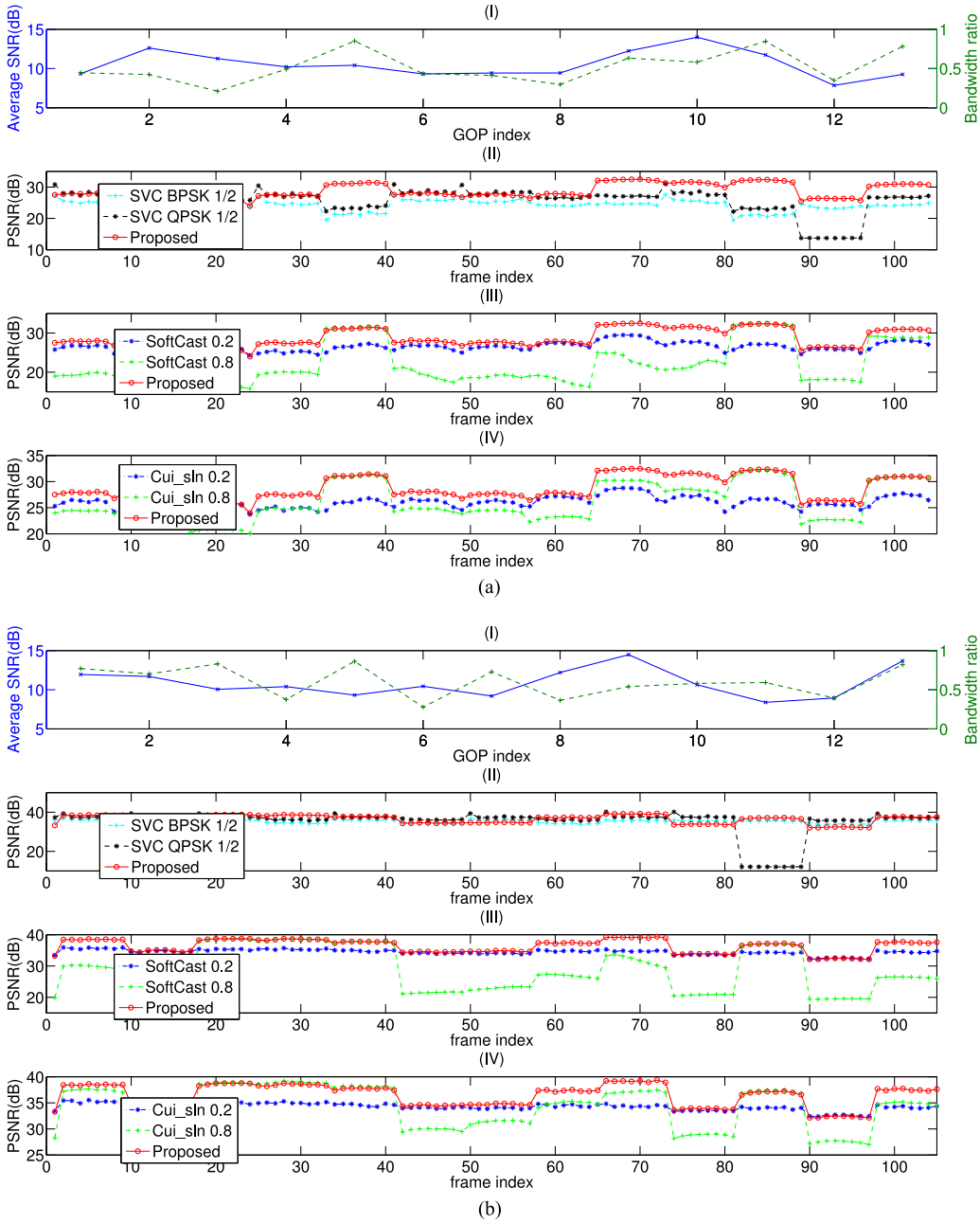


Fig. 7. Performance comparisons in dynamic environments. (a) *parkrun*. (b) *intotree*.

fading and noise trace obtained from the software radio platform SORA [8] in a mobile setting. The bandwidth ratio is simulated by uniformly varying between 0.2 and 0.9. Fig. 7(a)(I) and 7(b)(I) shows the bandwidth ratio and average channel SNR for each GOP. Three reference schemes are evaluated along with the proposed scheme. The PSNR of each frame is calculated to assess the channel SNR and bandwidth adaptation ability as well as the received video quality of different schemes. As examples, we show the results for sequences *parkrun* and *intotree* in Fig. 7.

As the available bandwidth varies between 20% to 90% of the full bandwidth, we implement a four layer SVC scheme, with each source layer using up to 20% of the full bandwidth. All four layers would take up to 80% of the full bandwidth. Since the traced channel SNR varies between 7 dB to 14 dB, we only consider BPSK and QPSK modulation. High-order modulations such as 16-QAM and 64-QAM would increase bandwidth efficiency, but would incur more transmission failures. Also, under such poor channel conditions, hierarchical modulation cannot be used. Therefore, we adopt a rate-1/2 convolutional code for all source layers. The source coding rate per layer when QPSK (BPSK) modulation is used is roughly 2.76 Mbps (1.38 Mbps). While there are other choices to implement an SVC scheme, it is not practical to evaluate all of them. The parameter settings in our implementation are general enough to draw a comparative conclusion between digital and analog schemes.

From subfigure (II) in Fig. 7(a) and 7(b), we see that scheme *SVC QPSK 1/2* performs better than *SVC BPSK 1/2* most of the time. This is obvious since using (QPSK 1/2) allows the sender to transmit the video stream at twice the bit rate of that when using (BPSK 1/2). However, notice that in the 12th GOP in the *parkrun* sequence and the 11th GOP in the *intotree* sequence, the performance of *SVC QPSK 1/2* decreases dramatically as the instantaneous channel SNR drops to around 8 dB. This is the typical *cliff effect* we often encounter in digital transmissions. If we had used an even higher coding and modulation rate (e.g. (16QAM 3/4)), such effect would have happen more often. There is always a trade-off in digital video streaming between high quality (high bit rate) and smooth experience (no sudden quality drop). Besides, there is also a trade-off between coding efficiency and scalability, so the coding efficiency of SVC is not as high as the standard H.264. In comparison, the *proposed* scheme smoothly adapts to both SNR and bandwidth variations and consistently achieves relatively high performance.

However, we can see from Fig. 7(b) that, in the 6th, 7th, 10th and 12th GOP, the performance of the proposed scheme is slightly inferior to the SVC-based digital scheme. In the 6th GOP, the bandwidth ratio is only 0.27. Such a low bandwidth ratio does not favor pseudo-analog transmission, because too many coefficients have to be discarded. The digital scheme, on the other hand, utilizes motion estimation and compensation for source de-correlation and achieves a high coding efficiency, so its performance under a low bandwidth ratio does not degrade too much. In fact, there exist approaches to improve the energy compaction efficiency of pseudo-analog transmission and in turn improve performance under low bandwidth ratio. Examples include the motion-compensated temporal filtering (MCTF) [19], [22] and hybrid digital analog (HDA) coding [3], [4]. We will leave the adoption of these techniques to our future work.

In the 7th GOP, the average channel SNR is slightly higher than 8 dB, which is the threshold SNR for (QPSK 1/2) to achieve error-free transmission. In the 10th and 12th GOP, the bandwidth is just enough to support 3 and 2 layers of the SVC stream. We can find that the digital scheme performs quite well when the (source and channel) coding and modulation scheme is matched to the channel condition. Therefore, in a stable wireless environment where both channel SNR and bandwidth provisioning can be precisely estimated, the digital solution is superior to the analog solution. However, if the channel changes dynamically and violently, an analog solution has unprecedented advantage over its digital counterpart.

We implement two variations for both pseudo-analog schemes (SoftCast and Cui_sln), with the target bandwidth ratios of 0.2 and 0.8. In the previous subsections, simulations have shown that these two pseudo-analog schemes have inherent SNR-adaptability, but cannot gracefully handle bandwidth variations. We can draw similar conclusions in the emulated environment, as shown in subfigures (III) and (IV) in Fig. 7(a) and 7(b). For both schemes, the implementation with a bandwidth ratio of 0.8 performs better when the bandwidth is sufficient, but performs much worse otherwise. The proposed progressive solution consistently outperforms SoftCast and Cui's solution for various channel conditions.

TABLE I
RUNNING TIME OF SCHEDULING A TILE

Time in Sec.	BW ratio 0.25	BW ratio 0.5	BW Ratio 1
real-time requirement	0.004	0.002	0.001
w/o speedup	0.008	0.013	0.015
w/speedup	0.002	0.001	0.001

D. Benchmark Evaluation

We first evaluate the running time of our scheduling and power allocation algorithm as described in Algorithm 1, whose worse case running time is $O(UK)$. We implement a single-thread version with C++ on a Windows 10 PC equipped with 3.4GHz Intel i7-3770 CPU. We set the GOP size to 8, and each frame consists of 256 chunks. There are 256 tiles to be scheduled in total. We perform ten independent runs and the average running time of scheduling a tile, denoted as t_s, is listed in Table I. In order to run in real-time, the following condition needs to be satisfied: $t_s \times 256 \times BWratio < 8/30$. We see from the table that the original algorithm is 2x to 15x slower than the real-time requirement. Although adopting multi-threading and more powerful hardware will make the algorithm run faster, we implement and test a speedup version of Algorithm 1 as follows: instead of scheduling a tile in the per-chunk granularity, it schedules a tile in the granularity of 2 successive chunks, 4 successive chunks and 8 successive chunks for the first 25%, the following 25% and the rest 50% tiles, respectively. This simple speed-up can satisfy the real-time requirement, as shown in Table I. The performance loss is small compared to the original version. For example, when the channel SNR is 15 dB, the performance degrades from 35.56 dB to 35.41 dB, 38.72 to 38.07 dB and 43.38 to 42.37 dB, when the bandwidth ratio is 0.25, 0.5 and 1, respectively. Therefore, our algorithm is ready for real-time implementation.

We also carry out an experiment to evaluate the impact of channel feedback delay d. For comparison, we set d to 0, 2 and $+\infty$ which means immediate feedback, feedback after 2 packets and no feedback, respectively. When the GOP has N chunks, the maximum number of packets to be transmitted for each GOP is $\Gamma = N/K$ (corresponding to the bandwidth ratio of 1). We carry out 200 test runs, and evaluate the performance when 25%, 50%, 75% and 100% of packets have been transmitted and received. Fig. 8 shows the average PSNR as well as the 10 and 90 percentile performances for the three delay settings.

It can be observed that, although the average PSNR degrades with the increased delay d, the performance loss is quite limited. In most of settings, the PSNR difference is less than 1 dB between cases where immediate feedback is available and no feedback is available. This experiment justifies our approximation proposed in Section IV-C and demonstrates that our scheme is applicable to scenarios where no channel feedback is available. Therefore, our solution is ready for multicast scenarios where channel feedback from each receiver is not usually possible.

Lastly, given that the input of the raw OFDM module of current digital 802.11 PHY has a limited bit length, representing

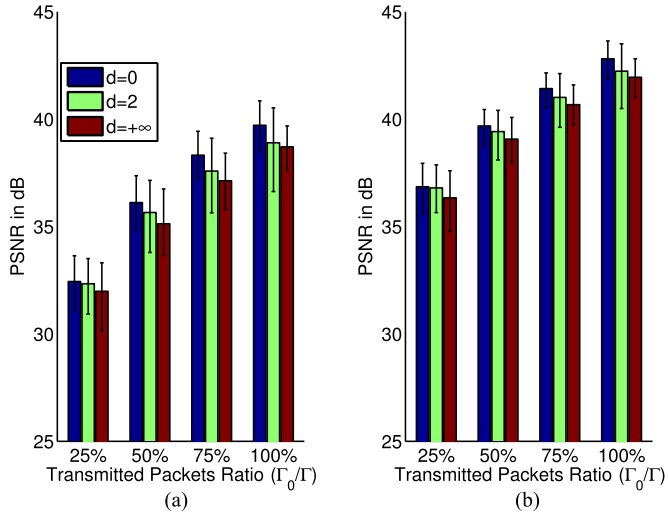


Fig. 8. Performance comparisons of difference schemes when channel SNR is 10 dB. The video sequences used are *City* and *Jets*. (a) *City*. (b) *Jets*.

TABLE II
AVERAGE RECEIVED PSNR WHEN CHANNEL SNR = 25 DB

	BW ratio 0.25	BW ratio 0.5	BW Ratio 1
Float OFDM input	37.0966 dB	41.0149 dB	52.0925 dB
16 Bits OFDM input	37.0964 dB	41.0147 dB	52.0914 dB

the analog coefficients in limited bit length will introduce some quantization error when integrating our progressive pseudo-analog transmission into current 802.11 PHY. Following the design of SORA [8], whose bit length of OFDM input is 16, we represent the absolute value of analog coefficients using the lowest 14 bits by quantization, the highest one is used as the sign bit and leaves the remaining bits reserved. We then evaluate the quantization loss over the 9 test videos. The smaller the channel noise power, the more difference the quantization loss will make. When channel SNR is as high as 25 dB, from Table II, we can see that quantization loss is negligible. Therefore, integrating pseudo-analog transmission into current 802.11 PHY is possible.

VI. CONCLUSION

In this paper, we propose a progressive pseudo-analog solution for bandwidth-adaptive and SNR-adaptive mobile video streaming. Through solving an optimization problem, we ensure that the receiver-side distortion is minimized after each packet is received. The solution for the optimal power allocation problem is derived and a low-complexity scheduling algorithm is presented. Evaluations in both simulated and real wireless environments show that the proposed scheme successfully achieves the design goal and outperforms state-of-the-art digital and pseudo-analog schemes by a notable margin in dramatically varying wireless environments.

REFERENCES

[1] "Cisco visual networking index: Global mobile data traffic forecast update 20142019 white paper," Cisco Systems, Inc., San Jose, CA, USA, Tech. Rep., 2015. [Online]. Available: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white_paper_c11-520862.pdf

[2] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[3] L. Yu, H. Li, and W. Li, "Wireless scalable video coding using a hybrid digital-analog scheme," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 2, pp. 331–345, Feb. 2014.

[4] D. He, C. Luo, C. Lan, F. Wu, and W. Zeng, "Structure-preserving hybrid digital-analog video delivery in wireless networks," *IEEE Trans. Multimedia*, vol. 17, no. 9, pp. 1658–1670, Sep. 2015.

[5] S. Jakubczak and D. Katabi, "A cross-layer design for scalable mobile video," in *Proc. 17th Annu. Int. Conf. Mobile Comput. Netw.*, 2011, pp. 289–300.

[6] S. Jakubczak, H. Rabul, and D. Katabi, "SoftCast: One video to serve all wireless receivers," Comput. Sci. Artif. Intell. Lab., Massachusetts Inst. Technol., Cambridge, MA, USA, Tech. Rep. MIT-CSAIL-TR-2009-005, 2009.

[7] C. Lan, D. He, C. Luo, F. Wu, and W. Zeng, "Progressive pseudo-analog transmission for mobile video live streaming," in *Proc. Vis. Commun. Image Process.*, Dec. 2015, pp. 1–4.

[8] K. Tan *et al.*, "Sora: High performance software radio using general purpose multi-core processors," in *Proc. 6th USENIX Network. Syst. Des. Implementation*, 2009, pp. 75–90.

[9] G. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[10] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.

[11] *ISO/IEC Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, ISO/IEC 8802-11 IEEE Std 802.11 Second edition 2005-08-01 ISO/IEC 8802 11:2005(E) IEEE Std 802.11i-2003 Edition, pp. 1–721, 2005.

[12] D. Astely *et al.*, "LTE: The evolution of mobile broadband," *IEEE Commun. Mag.*, vol. 47, no. 4, pp. 44–51, Apr. 2009.

[13] D. Rowitch and L. Milstein, "On the performance of hybrid FEC/ARQ systems using rate compatible punctured turbo (RCPT) codes," *IEEE Trans. Commun.*, vol. 48, no. 6, pp. 948–959, Jun. 2000.

[14] H. Cui, C. Luo, J. Wu, C. W. Chen, and F. Wu, "Compressive coded modulation for seamless rate adaptation," *IEEE Trans. Wireless Commun.*, vol. 12, no. 10, pp. 4892–4904, Oct. 2013.

[15] H. Choi, J. Yoo, J. Nam, D. Sim, and I. Bajic, "Pixel-wise unified rate-quantization model for multi-level rate control," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 1112–1123, Dec. 2013.

[16] S. Hu, H. Wang, S. Kwong, and T. Zhao, "Frame level rate control for H.264/AVC with novel rate-quantization model," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2010, pp. 226–231.

[17] T. Schierl, T. Stockhammer, and T. Wiegand, "Mobile video transmission using scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1204–1217, Sep. 2007.

[18] R. Xiong, F. Wu, J. Xu, and W. Gao, "Performance analysis of transform in uncoded wireless visual communication," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2013, pp. 1159–1162.

[19] H. Cui *et al.*, "Cactus: A hybrid digital-analog wireless video communication system," in *Proc. 16th ACM Int. Conf. Modeling Anal. Simul. Wireless Mobile Syst.*, 2013, pp. 273–278.

[20] R. Xiong *et al.*, "G-CAST: Gradient based image SoftCast for perception-friendly wireless visual communication," in *Proc. Data Compression Conf.*, Mar. 2014, pp. 133–142.

[21] X. L. Liu, W. Hu, Q. Pu, F. Wu, and Y. Zhang, "ParCast: Soft video delivery in MIMO-OFDM WLANs," in *Proc. Annu. Int. Conf. Mobile Comput. Netw.*, 2012, pp. 233–244.

[22] X. L. Liu *et al.*, "ParCast+: Parallel video unicast in MIMO-OFDM WLANs," *IEEE Trans. Multimedia*, vol. 16, no. 7, pp. 2038–2051, Nov. 2014.

[23] X. Zhao, H. Lu, C. Chen, and J. Wu, "Adaptive hybrid digital-analog video transmission in wireless fading channel," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 6, pp. 1117–1130, Jun. 2015.

[24] Z. Song, R. Xiong, S. Ma, X. Fan, and W. Gao, "Layered image/video softcast with hybrid digital-analog transmission for robust wireless visual communication," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2014, pp. 1–6.
[25] X. Fan, R. Xiong, D. Zhao, and F. Wu, "Layered soft video broadcast for heterogeneous receivers," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 11, pp. 1801–1814, Nov. 2015.
[26] H. Cui, C. Luo, C. W. Chen, and F. Wu, "Robust uncoded video transmission over wireless fast fading channel," in *Proc. IEEE INFOCOM'14*, Apr. 2014, pp. 73–81.

Dongliang He received the B.Sc. degree in electrical engineering and information science from the University of Science and Technology of China (USTC), Hefei, China, in 2012, and is currently working toward the Ph.D. degree in computer science at USTC.

His major research interests include multimedia communication and wireless networking.

Cuiling Lan received the B.Sc. degree in electrical engineering and the Ph.D. degree in intelligent information processing from Xidian University, Xi'an, China, in 2008 and 2014, respectively.

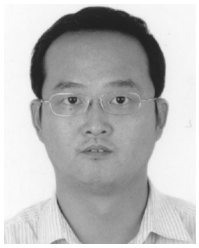
She joined Microsoft Research Asia, Beijing, China, in 2014. Her research interests include image and video compression, transmission, and computer vision.

Chong Luo (M'04–SM'14) received the B.Sc. degree from Fudan University, Shanghai, China, in 2000, the M.S. degree from the National University of Singapore, Singapore, in 2002, and the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China, in 2012.

She has been with Microsoft Research Asia, Beijing, China, since 2003, where she is currently a Lead Researcher with the Internet Media Group. Her research interests include wireless networking, wireless sensor networks, and multimedia communications.

Enhong Chen (M'06–SM'06) is currently a Professor, the Vice-Dean of the School of Computer Science, and the Vice-Director of the National Engineering Laboratory for Speech and Language Information Processing, University of Science and Technology of China, Hefei, China. He is also the Distinguished Young Scholar of the National Natural Science Foundation of China. He has authored and coauthorec more than 100 papers in refereed journals and conferences, including TKDE, TMC, TKDD, TIST, IJCAI, AAAI, KDD, ICDM, CIKM, and Nature Communications. His research interests include data mining and machine learning, social network analysis, and recommender systems.

Prof. Chen is a Fellow of the China Computer Federation. He is an Associate Editor of WWW Journal and the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS. He was on program committees of numerous conferences including IJCAI, AAAI, KDD, ICDM, and SDM. He was the recipient of the Best Application Paper Award on KDD'08, the Best Research Paper Award on ICDM'11, and the Best of SDM'15 Award.



Feng Wu (M'00–SM'06–F'13) received the B.S. degree in electrical engineering from Xidian University, Xi'an, China, in 1992, and the M.S. and Ph.D. degrees in computer science from Harbin Institute of Technology, Harbin, China, in 1996 and 1999, respectively.

He is currently a Professor with the University of Science and Technology of China, Hefei, China. Before that, he was Principle Researcher and Research Manager with Microsoft Research Asia, Beijing, China. He has authored or coauthored more than 200 high-quality papers (including several dozens IEEE Transaction papers) and top conference papers on MOBICOM, SIGIR, CVPR, and ACM MM. He has 77 granted U.S. patents. His 15 techniques have been adopted into international video coding standards. His research interests include image and video compression, media communication, and media analysis and synthesis.

Prof. Wu was an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEM FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON MULTIMEDIA, and several other international journals. He was the recipient of the IEEE Circuits and Systems Society 2012 Best Associate Editor Award. He was also the TPC Chair in MMSP 2011, VCIP 2010, and PCM 2009, and Special Sessions Chair in ICME 2010 and ISCAS 2013. As a coauthor, he was a recipient of the Best Paper Award in the IEEE T-CSVT 2009, PCM 2008, 1121 and SPIE VCIP 2007.

Wenjun (Kevin) Zeng (M'97–SM'03–F'12) received the B.E. degree from Tsinghua University, Beijing, China, in 1990, the M.S. degree from the University of Notre Dame, Notre Dame, IN, USA, in 1993, and the Ph.D. degree from Princeton University, Princeton, NJ, USA, in 1997.

He is currently a Principal Research Manager with Microsoft Research Asia, Beijing, China, and a Full Professor with the Department of Computer Science, University of Missouri (MU), Columbia, MO, USA. Prior to joining MU in 2003, he worked for PacketVideo Corporation, San Diego, CA, USA; Sharp Labs of America, Camas, WA, USA; Bell Labs, Murray Hill, NJ, USA; and Panasonic Technology, Princeton, NJ, USA. He has contributed significantly to the development of international standards (ISO MPEG, JPEG2000, and OMA), and has developed wireless video streaming products that have been widely used. His current research interest includes mobile-cloud video analysis, social network/media analysis, multimedia communications/networking, and content/network security.

Prof. Zeng is/was an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (TCSVT), the *IEEE MultiMedia Magazine* (currently an Associate EiC), the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, and the IEEE TRANSACTIONS ON MULTIMEDIA (TMM), and is/was on the Steering Committee of the IEEE TRANSACTIONS ON MOBILE COMPUTING(current) and the IEEE TMM (2009–2012). He was the Steering Committee Chair of the IEEE International Conference Multimedia and Expo in 2010 and 2011, and was the TPC Chair/Co-Chair of several IEEE conferences (e.g., ChinaSIP'15, WIFS'13, ICME'09, and CCNC'07). He will be a General Co-Chair of ICME2018. He is currently guest editing a TCSVT Special Issue on Visual Computing in the Cloud—Mobile Computing, and was a Guest Editor (GE) of ACM TOMCCAP Special Issue on ACM MM 2012 Best Papers, a GE of the PROCEEDINGS OF THE IEEE Special Issue on Recent Advances in Distributed Multimedia Communications (January 2008), and the Lead GE of the IEEE TMMs Special Issue on Streaming Media (April 2004).