A comprehensive theoretical evaluation of the end-to-end performance of SoftCast-based linear video delivery schemes

Anthony Trioux, M Gharbi, François-Xavier Coudoux, Patrick Corlay

A comprehensive theoretical evaluation of the end-to-end performance of SoftCast-based linear video delivery schemes

Anthony Trioux, M Gharbi, François-Xavier Coudoux, Patrick Corlay

A Comprehensive Theoretical Evaluation of the End-to-End Performances of SoftCast-based Linear Video Delivery Schemes

Anthony Trioux, Mohamed Gharbi, François-Xavier Coudoux, *Senior Member, IEEE,* and Patrick Corlay

***Abstract*—SoftCast-based linear video delivery (LVD) schemes have been proposed as an alternative to traditional video transmission schemes in wireless error-prone environments. The end-to-end performance of SoftCast-based schemes have been evaluated in [1], where a theoretical model based on the Peak Signal-to-Noise Ratio (PSNR) metric has been proposed. The latter is limited to the use of a Zero-Forcing (ZF) estimator at the receiver side, and does not consider bandwidth restrictions. Nevertheless, bandwidth restrictions are common and necessary in practice, especially when considering the transmission of video content. It is mandatory to take this aspect into consideration as it may drastically influence the received video quality. In this paper, we provide valid and significant extensions of the initial model. In total, three models are introduced taking into account both 1) bandwidth constraints (*i.e.*, data compression applied), 2) the use of a Linear Least Square Error (LLSE) estimator instead of the ZF one as well as 3) the use of the optimal power allocation. We show that regardless of the bandwidth reduction applied, the type of estimator as well as the power allocation used, the end-to-end video quality can be accurately modeled and predicted at the transmitter according to the video content characteristics, the type of estimator used at the receiver and the channel conditions. The validity of these three models is assessed through extensive end-to-end simulations. These new models give solid theoretical guidelines for optimizing and studying the performance of linear video delivery schemes.**

***Index Terms*—Wireless Video Transmission, SoftCast, Uncoded Transmission, Linear Video Delivery, Distortion Model, Bandwidth Constraints**

I. Introduction

VIDEO transmission to and from mobile users is an increasingly relevant service for current and next generation wireless networks. According to Cisco Visual Networking Index report, about 80% of the world's mobile data traffic will be video by 2022 [2]. As a consequence, a huge research effort is devoted to designing video coding and transmission systems that give the best video quality for a given amount of wireless resources. This is especially difficult when considering the following cases: (i) the video has to be sent to many mobile users, each experiencing different channel characteristics; (ii) the channel characteristics change quickly over time. In severely delay-constrained video applications such as videoconferencing, telepresence, and teleoperation, ultra-low latency video coding and transmission becomes even more challenging.

Currently, to perform video transmission, traditional video coders (e.g., H.264/AVC [3], HEVC [4]) are usually combined with a transmission scheme over a suitable network protocol (e.g., 802.11, 4G). Even though this solution works well for stable point-to-point communications, it is not suitable when considering multiple receivers, broadcast context, and wireless environments where the channel quality may vary over time. Indeed, it suffers from some inherent limitations:

- First, the source and channel coding parameters have to be decided by the transmitter and are the same for all the receivers. However, due to unreliable wireless channels that vary over time, receivers can experience cliff effect [5] or leveling-off effect [6]. The cliff effect refers to the fact that the video quality drops quickly (due to glitches or freeze of the video) when the channel quality is below a presumed value. The leveling-off effect refers to the fact that the received video quality remains constant even when the channel quality keeps increasing;
- Second, to accommodate channel quality fluctuations, traditional techniques use rate control mechanisms combined with adaptive modulation and coding (AMC) [7] mechanisms, which require a permanent adaptation of the coding parameters by the transmitter. These techniques rely on the estimation of the rate-distortion characteristic of the source and the channel characteristics [7], implying additional delay to perform this adaptation. Furthermore, such mechanisms are often designed for unicast applications but become difficult to apply when considering multiple receivers;
- Third, delay is also caused by various buffers present at the encoder, within the network, and at the receiver [8], [9]. Buffers are used to smooth out variations of the channel characteristics and video coder rate. They are also necessary due to the shared network infrastructure;
- Finally, when the errors in a received packet cannot be recovered by the Forward Error Correction (FEC) codes, the entire packet is usually discarded and retransmitted if possible using Automatic Repeat reQuest (ARQ) mechanism [6]. Retransmissions induce again significant additional delays [10], which is incompatible with low-latency use-cases. However, if the retransmission mechanism is not used, the video quality drops quickly.

To address some of these issues, one may use instead scalable video coders (e.g., H.264/SVC [11], SHVC [12]) combined with hierarchical modulation (HM) [13], which deliver multiple signals: first, a base layer with low bitrate strongly protected and then, one or multiple enhancement layer(s) to increase the reconstructed quality. The aim of this mechanism is to maximize the number of users that can decode the video. Since several layers are available, this could mitigate the channel variation adaptation problem. However, it is well known that scalable coding suffers from compression inefficiency (roughly +15% to +25% additional rate per layer

The authors are with the Institute of Electronics, Microelectronics, and Nanotechnology (UMR CNRS 8520), Department OAE–Hauts-de-France Polytechnical University, Le Mont Houy, Valenciennes, 59313, France (e-mail: anthonytrioux@laposte.net).

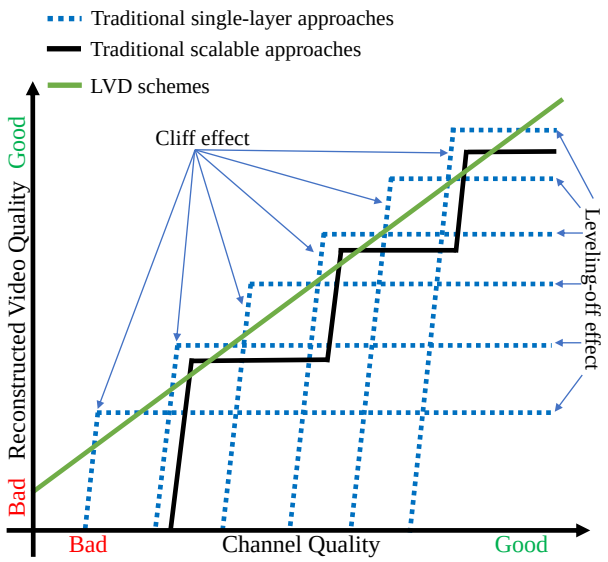


Fig. 1. Illustration of the performance of the traditional video coders in wireless transmission vs. LVD schemes [14], [21]–[23].

[12]), additional complexity and additional delay (an encoder instance must be run per layer, often in a sequential fashion). Furthermore, the cliff-effect obtained with the single-layer solution is just divided into a few smaller cliffs [14] without eliminating it completely as shown in Fig. 1.

In the last few years, the so-called *SoftCast* scheme [15] has emerged and demonstrated a high potential to address some of the aforementioned issues. SoftCast represents the pioneer work of linear video delivery (LVD) schemes [16] also known as soft video delivery [17] or uncoded video transmission [1]. All of these designations stand for the fact that video pixels are processed by successive linear operations and are directly transmitted in a pseudo-analog way, *i.e.*, without either quantization or entropy and channel coding. This allows to:

- Deliver to each receiver a video quality that is a linear function of the user's channel quality[1] [1], [18];
- Achieve graceful degradation and avoid cliff effect [19] caused by traditional approaches as shown in Fig. 1;
- Work without any feedback from receivers [15];
- Send only a single data stream, which is decodable by any receiver, even those experiencing bad channel conditions [20]. Therefore, rate control and AMC are no longer needed [7]. Furthermore, since all the packets corrupted by noise can be decoded, retransmission procedures are also avoided. The channel noise simply corrupts the amplitude of the transmitted (DCT) coefficients. Visually, it corresponds to a snow-effect, which is illustrated in Fig. 2b. The more the noise, the more visible the snow-effect [18]. However, the video remains clearly viewable in contrast to traditional approaches that suffer from glitches as shown in Fig. 2c.

Moreover, LVD systems offer a relatively low and controlled latency [8] that can be adjusted through the size of the temporal transform. They have the potential of dramatically improving the quality of experience in wireless and latency-constrained scenarios, which represent a paradigm break with respect to traditional video transmission systems.

[1]As shown in this paper, the linear relationship is no longer valid over the entire channel quality range when considering bandwidth restrictions.

(a) Original Image

(b) PSNR=34.97dB

(c) PSNR=15.04dB

Fig. 2. Visual quality comparison at a channel quality = 7dB for the *Intotree* video sequence (#1 frame), no compression applied. (a) Original image, (b) SoftCast, (c) HEVC-compressed video content transmitted with BPSK modulation.

For all these reasons, LVD schemes have recently gathered a significant interest from the research community [1], [6]–[8], [14]–[40]. Works concern for instance the use of different decorrelation transforms e.g., Discrete Wavelet Transform (DWT) [24], the human visual system properties [20], [25] and the bandwidth-constrained environments [6], [26]. Furthermore, their comparisons with state-of-the-art traditional video coding schemes, as well as guidelines for real implementations can be found in the literature, showing the potential and applicability of the LVD schemes. For instance, Garuffa *et al.* [29] provided a detailed comparison between the HEVC video coding scheme and SoftCast using SDR modules. They showed that SoftCast offers better quality than HEVC when considering multicast applications as well as when the channel quality is very low, since HEVC cannot be correctly decoded. In a more recent paper, Tang *et al.* [30] proposed guidelines to implement the SoftCast scheme on Software Defined Radio (SDR) modules using the GNU Radio software. They found that the difference between PSNR scores obtained in simulations and those obtained in real experiments is only about 0.25dB, which is visually unnoticeable [41].

Among the existing works, Xiong *et al.* [1] proposed a theoretical analysis of the LVD schemes. The authors showed that the Peak Signal-to-Noise Ratio (PSNR) metric, used as a measure of the video quality, is a linear function of the channel quality, hereafter measured through the Channel Signal-to-Noise Ratio (CSNR). Nevertheless, their model only considers the case where the video signal can be fully transmitted, *i.e.*, without applying any compression. Furthermore, only the Zero-Forcing (ZF) estimator is taken into account. This model is unusable when considering practical bandwidth-constrained applications since the linear relationship between video quality and channel quality is broken, as shown in this paper.

The purpose of this paper is to generalize and to provide valid and significant extensions of the model proposed by Xiong *et al.* [1] by taking into account both bandwidth constraints and different types of estimator used at the receiver. In addition, an original model considering the optimal power allocation [16], [31], [32] in a SoftCast-based scheme is also investigated. This study leads to three theoretical models, which perfectly match all the simulation configurations (different available bandwidths, GoP-sizes, channel qualities, etc.). Hence, the proposed research gives solid theoretical guidelines for studying and optimizing LVD schemes. Specifically, the proposed models are useful to:

1) Provide a better understanding of SoftCast-based video delivery and a faster evaluation/comparison with future LVD methods developed by the research community;
2) Study the performance of the LVD schemes and quickly optimize their parameters (e.g., GoP-size, chunk-size, etc.) without having to perform extensive end-to-end simulations;
3) Estimate, in real conditions at the receiver side, the PSNR score of the reconstructed video without having the original video. Indeed, the original video is usually necessary to compute the PSNR metric, but in practice, it is not available at the receiver side. Since our models rely on the power distribution characteristics, which is sent by the LVD schemes to the receiver, it is possible to compute the PSNR score without having the original video.

The rest of this paper is organized as follows: Section II provides background on the SoftCast scheme. Section III introduces the theoretical analysis of the end-to-end performance for SoftCast-based schemes using the ZF estimator. Section IV gives the theoretical models of SoftCast-based schemes considering the LLSE estimator. Section V presents the theoretical model of SoftCast-based schemes using optimal power allocation. In section VI, the proposed models are compared together to study the performance of the different schemes. Furthermore, the usefulness of the models is demonstrated by studying the performance of the schemes according to different GoP-sizes. Conclusions are given in section VII.

II. SoftCast Overview

The basic scheme of SoftCast [15] is illustrated in Fig. 3. SoftCast first takes a Group of Pictures (GoP) and uses a full-frame 3D-DCT as a decorrelation transform. The DCT-frames

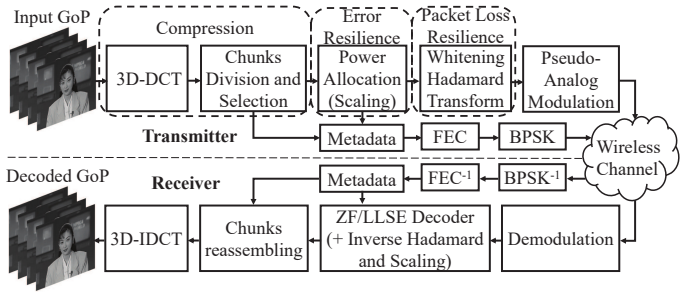


Fig. 3. Block diagram of the SoftCast video transmission scheme.

are then divided into N small rectangular blocks of transformed coefficients called *chunks*. These chunks are rearranged by decreasing order according to their energy denoted by $\lambda_i = E[\boldsymbol{x_i^2}], i = 1, 2, \ldots, N$. In the SoftCast scheme, the data compression can be done after the decorrelation transform. Specifically, when B, the available channel bandwidth for the transmission is less than the signal bandwidth, *i.e.*, only the $K \leq N$ chunks may be transmitted, SoftCast discards the $N - K$ chunks with less energy. This is generally the case especially for the transmission of High Definition (HD) content as mentioned in [6], [25], [33]. At the receiver side, these discarded chunks are replaced by null values [15]. To represent the bandwidth limitation, we introduce the compression ratio [33] (CR) defined as:

$$\text{CR} = \frac{K}{N} \tag{1}$$

When CR=1, no compression is applied. Likewise, when CR=0.25, only 25% of the chunks are transmitted [33].

In parallel, the SoftCast transmitter also sends side information referred to as *metadata*. Metadata consist of the mean and the variance of each transmitted chunk as well as a binary map, which indicates the positions of the discarded chunks into the GoP. They are strongly protected and transmitted in a robust way (e.g., BPSK [17]) to ensure error-free decoding. Therefore, B is split into two parts B_1 and B_2 [17]: the bandwidth for metadata transmission and the bandwidth to transmit the video. In practice, the transmission of the metadata may require to discard more chunks at the transmitter.

The third block at the transmitter level called Power Allocation or Scaling is used to provide error resilience. SoftCast scales the magnitude of the DCT coefficients to offer a better protection against transmission noise. Since the total transmission power available P is limited and fixed, it is distributed to all the chunks in a way that minimizes the Mean Square reconstruction Error (MSE) between transmitted and decoded chunks. This is a typical Lagrangian problem which offers two solutions depending on whether or not the channel noise power is known by the transmitter. In the first case, *i.e.*, when the channel noise power is unknown by the transmitter, it leads to the following near-optimal solution [15], [17] given by:

$$g_i = \lambda_i^{-1/4} \cdot \sqrt{\frac{P}{\sum_i \sqrt{\lambda_i}}}, \tag{2}$$

where $g_i, i = 1, 2, \ldots, N$ is the scaling factor for the ith chunk.

In the second case, *i.e.*, when the channel noise power is known by the transmitter, the optimal power allocation can be performed. In this case, only the $\ell \leq N$ chunks with power level above the noise power level (σ_n^2) are transmitted. This scheme is known as SoftCast+ [34] and the corresponding solution [16], [31], [32] is given by:

$$g_m = \frac{\left(\sqrt{\frac{\lambda_m \sigma_n^2}{\gamma}} - \sigma_n^2\right)^{1/2}}{\sqrt{\lambda_m}}, \tag{3}$$

where $g_m, m = 1, \ldots, \ell$ is the optimal scaling factor for the mth chunk, and:

$$\sqrt{\gamma} = \frac{\sigma_n \sum_{m=1}^{\ell} \sqrt{\lambda_m}}{P + \ell {\sigma_n}^2}. \tag{4}$$

Note that only one scaling factor per chunk is computed in SoftCast, which results from a trade-off among amount of metadata, received quality and computation cost. Readers may refer to [15] for further details.

After the power allocation, the Hadamard transform is applied to the scaled coefficients to provide packet loss resilience. This process transforms the chunks into *slices*. Each slice is a linear combination of all scaled-chunks. Finally, these packets are transmitted in a pseudo-analog manner using Raw-OFDM [35], *i.e.*, classical coding (e.g., FEC code) and digital modulation stages are skipped [15].

At the receiver side, if the channel noise power is unknown, the ZF estimator is used to recover the transmitted symbols as in [1]. Otherwise and as commonly used in the original works of Jakubczak *et al.* [15] or in [16]–[20], [24]–[28], [30]–[37], the LLSE estimator is preferred to minimize the impact of the noise on the received symbols.

Using the metadata, the decoded values are then reassembled to form DCT-frames, which are then passed through an inverse 3D-DCT process.

To make the paper easier to understand, we introduce in the following the general SoftCast-based transmission chain in Fig. 4, where α_i is the decoding factor at the receiver and represents either the use of the ZF or LLSE estimator. Likewise, β_i is the scaling factor at the transmitter and represents either the use of the near-optimal power allocation as classically used in SoftCast [15], [17], [33] or the optimal power allocation [16], [31], [32]. The index i indicates that the process is done chunk by chunk.

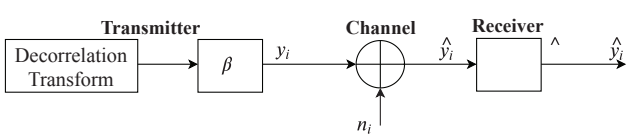


Fig. 4. General SoftCast-based transmission chain.

All the herein proposed models are based on the assumption that the channel is an Additive White Gaussian Noise (AWGN) one. Although this type of channel does not fully represent what can happen in wireless environments (fading, burst errors, etc.), this assumption is often used in a SoftCast context [1], [14], [17], [18], [20], [21], [24]–[27], [29]–[33], [37]–[40]. The main reason is given below [38]. First, we remind that SoftCast transforms chunks into slices using the Hadamard transform. The latter acts as a whitening data process by ensuring that each slice carries approximately the same energy. In reality, when these slices are transmitted over the channel, they can experience different fading. At the receiver level, they are divided by their respective fading coefficient which implies that the noise power distribution is not homogeneous over the data. However, after applying the inverse Hadamard transform, the noise power is redistributed and whitened over all the chunks, which can be approximated by an AWGN channel.

Note that the Hadamard transform is not considered in the following analysis as it does not change either the transmission power or channel noise characteristics. Indeed, the Hadamard transform is an orthogonal transform. Furthermore, if n represents an Additive White Gaussian Noise, then $n' = H^{-1} \cdot n$ is also AWGN. Proofs can be found in [1].

Furthermore, recall that uncoded video transmission uses a linear decorrelation transform step such as DCT [15] or DWT [24]. All the developments below, including the proposed theoretical models are valid regardless of the used linear transform (e.g., 2D-DCT for images, 3D-DCT or 3D-DWT for videos, etc.).

In the rest of this paper and for clarity, we denote:

- the SoftCast-based scheme with ZF decoder and near-optimal power allocation by SoftCast(ZF);
- the SoftCast-based scheme with LLSE decoder and near-optimal power allocation by SoftCast(LLSE);
- the SoftCast-based scheme with optimal power allocation and LLSE estimator by SoftCast+;
- the case where all the coefficients can be transmitted by the FB acronym (Full Bandwidth, *i.e.*, no bandwidth restriction), corresponding to CR=1;
- the Constrained-Bandwidth applications by the CB acronym, corresponding to CR<1.

In the next section, we first recall Xiong's model [1] and then introduce the new model that takes into account bandwidth constraints environments.

III. Description of the theoretical models considering ZF estimator

A. Background (Xiong's Model)

Xiong *et al.* considered a full bandwidth (FB) application case where an arbitrary vector $\mathbf{x} = (x_1, x_2, \ldots, x_N) \in \mathbb{R}^N$ is transmitted over an AWGN channel. The N elements $\{x_i\}$, $i = 1, 2, \ldots, N$ represent either pixels or coefficient values after orthogonal transform. The corresponding received vector is denoted by $\hat{\mathbf{x}}$.

As explained in Section II, the sender scales the x_i before transmitting them:

$$y_i = \beta_i \cdot x_i, \tag{5}$$

where $\beta_i = g_i$, with g_i given by (2).

The channel type considered by Xiong *et al.* as well as the one in this paper is AWGN, *i.e.*, the transmitted signal is contaminated by additive white Gaussian noise:

$$\begin{aligned}\hat{y}_i &= y_i + n_i, \\ &= g_i \cdot x_i + n_i.\end{aligned} \tag{6}$$

where n_i is the channel noise.

Assuming the use of the ZF estimator, the estimated elements of $\hat{\mathbf{x}}$ at the receiver side are given by [1]:

$$\begin{aligned}\hat{x}_i &= \hat{y}_i \cdot \alpha_i \\ &= \frac{\hat{y}_i}{g_i} \\ &= x_i + \frac{n_i}{g_i}.\end{aligned} \tag{7}$$

The expected distortion in $\hat{x_i}$ is:

$$\begin{aligned}D_i &= E\big[(\hat{x}_i - x_i)^2\big], \\ &= \frac{E\big[n_i^2\big]}{g_i^2}, \\ &= \frac{\sigma_n^2}{g_i^2}.\end{aligned} \tag{8}$$

Here σ_n^2 is the power of the channel noise. The expected transmission power for sending x_i is:

$$\begin{aligned}P_i &= E\big[{y_i}^2\big], \\ &= g_i^2 \cdot E\big[{x_i}^2\big].\end{aligned} \tag{9}$$

Combining (8) and (9), we get the power-distortion function of uncoded transmission:

$$D_i \cdot P_i = \sigma_n^2 \cdot \lambda_i,$$

or

$$D_i = \frac{\sigma_n^2}{P_i} \cdot \lambda_i. \tag{10}$$

To achieve optimal performance considering the ZF estimator, the total transmission power available at the transmitter P, should be allocated among all the elements x_i by:

$$(P1) : \min \sum_{i=1}^{N} D_i, \text{s.t.} \sum_{i=1}^{N} P_i \leq P \tag{11}$$

This is a Lagrangian problem:

$$\mathcal{L} = \sum_{i=1}^{N} D_i + \frac{1}{C_{\text{ZF}}^2} \sum_{i=1}^{N} P_i, \tag{12}$$

where C_{ZF}^2 is the Lagrange multiplier.

Differentiating (13) with respect to P_i and setting the result to zero, we get:

$$C_{\text{ZF}}^2 = \frac{P_i^2}{\lambda_i \sigma_n^2}. \tag{13}$$

This determines the optimal power for sending x_i:

$$P_i = C_{\text{ZF}}^2 \sigma_n \sqrt{\lambda_i}. \tag{14}$$

Since there exists a constraint on the total power transmission:

$$\sum P_i = P, \tag{15}$$

C_{ZF} can be obtained as:

$$C_{\text{ZF}} = \frac{P}{\sigma_n \sum \sqrt{\lambda_i}}. \tag{16}$$

Recall the equations (10) and (14), we easily get:

$$D_i = \frac{\sigma_n^2}{P_i} \lambda_i = \frac{\sigma_n}{C_{\text{ZF}}} \sqrt{\lambda_i}. \tag{17}$$

Therefore, recalling that the sender transmits all the N elements of $\mathbf{x}$ (*i.e.*, full bandwidth is available at the transmitter) and that the ZF estimator is used at the receiver side, the total expected distortion denoted by $\text{D}_{\text{[ZF/FB]}}$ under optimal power allocation for uncoded transmission is given by:

$$\text{D}_{\text{[ZF/FB]}} = \sum_{i=1}^{N} D_i = \frac{\sigma_n^2}{P} \left(\sum_{i=1}^{N} \sqrt{\lambda_i} \right)^2, \tag{18}$$

recalling that σ_n^2 is the noise power and λ_i the energy of the ith transmitted element of $\mathbf{x}$ [15].

Based on the following definition of the CSNR and PSNR expressed in decibels:

$$\text{CSNR} = 10\log_{10}(\bar{P}/\sigma_n^2), \quad \bar{P} = P/N, \tag{19}$$

$$\text{PSNR} = 10\log_{10}(255^2/\bar{D}), \quad \bar{D} = \text{D}/N. \tag{20}$$

They showed that the expected reconstructed video quality without data compression can be finally obtained from:

$$\text{PSNR}_{\text{[ZF/FB]}} = c + \text{CSNR} - 20\log_{10}\big(H\big), \tag{21}$$

where $c = 20\log_{10}(255)$ and

$$H = \frac{1}{N} \sum_{i=1}^{N} \sqrt{\lambda_i}, \tag{22}$$

represents the *data activity* [1]. The higher the data activity H, the lower the reconstructed PSNR, showing the importance of taking into account the characteristics of the transmitted video content in a SoftCast context [1], [33]. Note the linear characteristic of the $\text{PSNR}_{\text{[ZF/FB]}}$ that depends on the channel transmission conditions.

B. The proposed model for constrained-bandwidth (CB) applications

We propose to extend the study by considering the more realistic and general case *i.e.*, only the $K \leq N$ largest energy elements are transmitted due to bandwidth constraints. Thus, the total distortion $\text{D}_{\text{[ZF/CB]}}$ now consists of two parts:

- The distortion D_i due to transmission, which now affects K transmitted elements (x_i) instead of N. For ease of reading, let us denote the overall distortion due to transmission $D_s = \sum_{i=1}^{K} D_i$, where $D_i = E\big[(\hat{x}_i - x_i)^2\big]$.
- The distortion D_j due to compression, coming from each of the $N - K$ discarded elements (x_j). Indeed, since these elements are not sent at all, the information that they carry cannot be recovered at the receiver side. Mathematically, the corresponding distortion is given by: $D_j = E\big[(0 - x_j)^2\big]$, as if these elements were received

with a null value. Likewise, we denote the overall distortion due to compression $D_d = \sum_{j=K+1}^{N} D_j$.

Therefore, in this case, the total distortion (18) observed at the receiver is rewritten as the sum of these two terms:

$$\begin{aligned} \mathrm{D}_{[\mathrm{ZF/CB}]} &= D_s + D_d, \\ &= \frac{\sigma_n^2}{P}\left(\sum_{i=1}^{K}\sqrt{\lambda_i}\right)^2 + \sum_{j=K+1}^{N}\lambda_j. \end{aligned} \tag{23}$$

We note that the average transmission power in (19) becomes $\bar{P} = P/K$ as the total transmission power is here distributed over the K transmitted elements of $\mathbf{x}$.

By inserting (23) into (20), we get:

$$\begin{aligned} \mathrm{PSNR}_{[\mathrm{ZF/CB}]} &= 10\log_{10}\left(\frac{255^2\cdot N}{D_s + D_d}\right), \\ &= c - 10\log_{10}\left(1+\frac{D_d}{D_s}\right) + 10\log_{10}\left(\frac{\bar{P}}{\sigma_n^2}\right) \\ &\quad - 10\log_{10}\left(\frac{1}{NK}\left(\sum_{i=1}^{K}\sqrt{\lambda_i}\right)^2\right). \end{aligned} \tag{24}$$

By analogy with (21), we identify the new data activity of the transmitted elements as:

$$H_t = \frac{1}{\sqrt{NK}}\sum_{i=1}^{K}\sqrt{\lambda_i}. \tag{25}$$

For ease of reading, we also define E_d, the overall energy of all discarded elements:

$$E_d = \frac{1}{N}\sum_{j=K+1}^{N}\lambda_j. \tag{26}$$

According to these new definitions, the end-to-end video quality considering bandwidth constraints for the ZF estimator is finally given by:

$$\begin{aligned} \mathrm{PSNR}_{[\mathrm{ZF/CB}]} &= c + \mathrm{CSNR} - 20\log_{10}\left(H_t\right) \\ &\quad - 10\log_{10}\left(1+\frac{\mathrm{CSNR}_{lin}\cdot E_d}{{H_t}^2}\right). \end{aligned} \tag{27}$$

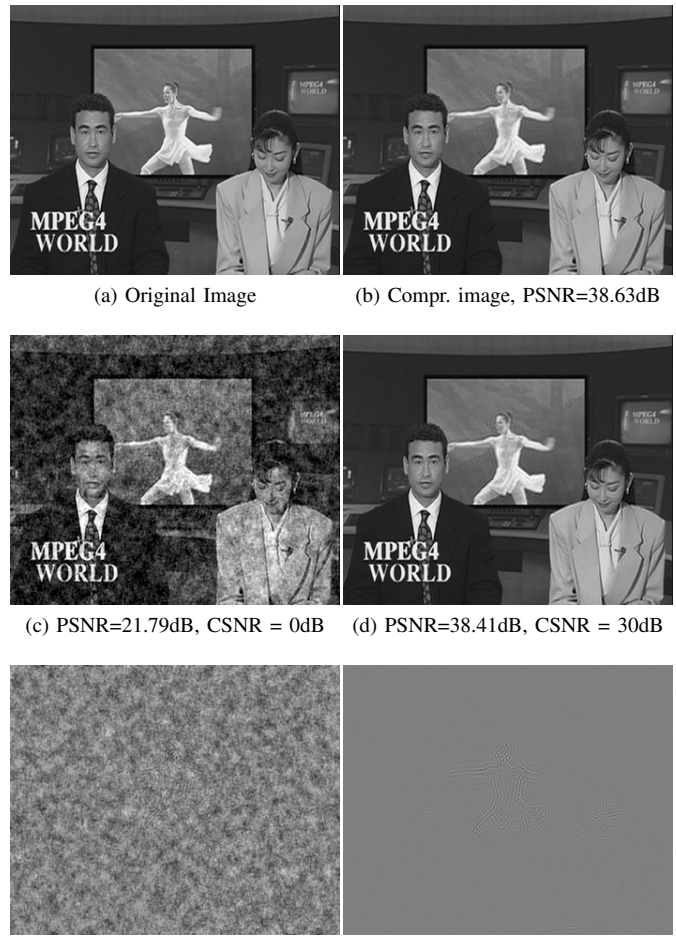
where $\mathrm{CSNR}_{lin} = \frac{\bar{P}}{\sigma_n^2}$.

The above equation includes a new term in comparison to (21) that reflects the effect of the data compression applied. The PSNR now depends on three parameters: the CSNR, H_t and E_d. The CSNR depends on the transmission conditions, whereas the two other terms depend on the energy of the transmitted and discarded elements. In practical applications, it is therefore possible to estimate the PSNR score of the reconstructed video at the receiver side without needing the original video. Indeed, the value of the CSNR can be estimated through the pilot symbols and the value of H_t can be computed through the transmitted metadata. For E_d, since the elements are discarded, the value of each λ_j is normally unknown at the receiver but it can be transmitted as a unique additional metadata with a low cost (e.g., 32 bits per GoP).

For a given bandwidth, the higher E_d, the greater degradation. However, as E_d is multiplied by the CSNR_{lin}, the degradation coming from the compression becomes less noticeable at low CSNR environments. We give in the following an intuitive explanation and illustrate this effect in Fig. 5. We first fix the compression level to CR $= 0.25$ (75% of the elements/chunks are discarded). Then we consider two cases:

- First, the channel quality is bad (e.g., CSNR=0dB). In such cases, SoftCast images suffer from high distortion (snow effect) coming from the transmission errors as shown in Fig. 5c and Fig. 5e. This distortion due to transmission is predominant and acts as a masking effect letting the distortion due to compression almost unnoticeable;
- In contrast, when the channel quality is good (e.g., CSNR=30dB), the distortion due to the transmission tends to null value (the snow effect disappears) as shown in Fig. 5d and Fig. 5f. Therefore, the degradation due to the compression now becomes clearly visible (e.g., ringing artifacts on the ballerina in the foreground as well as on the female presenter).

When $K = N$, *i.e.*, CR=1, (27) and (21) are identical. In



(a) Original Image (b) Compr. image, PSNR=38.63dB

(c) PSNR=21.79dB, CSNR = 0dB (d) PSNR=38.41dB, CSNR = 30dB

(e) PSNR=21.79dB, CSNR = 0dB (f) PSNR=38.41dB, CSNR = 30dB

Fig. 5. Visual quality comparison at a CR=0.25 for *Mixed CIF* sequence (#225 frame), GoP-size=16 frames. (a) Original image, (b) Compressed image, (c) SoftCast(ZF) with a CSNR = 0dB, (d) SoftCast(ZF) with a CSNR = 30dB. (e), (f) Resulting error images for CSNR=0dB and CSNR=30dB, respectively. Please enlarge the figure to observe details.

other words, the video quality scales linearly with the CSNR as stated in [1].

The effectiveness of the proposed model is compared to the original SoftCast scheme. Therefore, without loss of generality for (27), the elements of $\mathbf{x}$ now represent chunks and all the linear operations (3D-DCT, scaling, etc.) are implemented in compliance with [15]. However, for this section the ZF estimator is used instead of the LLSE one. The model for such an estimator is given in the next section.

The model is evaluated through extensive simulations using MATLAB® by considering several video contents: User Generated Content (UGC) with 360p, 480p and 1080p resolutions from the recent YouTube UGC database [42] as well as traditional HD720p sequences (class E, 1280×720 pixels, 30fps) and CIF sequences (352×288 pixels, 30fps) from the Xiph collection [43] or from the JCT-VC database [44]. To summarize the results, we create two different *Mixed* sequences by slicing the first 128 frames of ten sequences from [43] or [44]. First, the *Mixed HD720p* sequence, composed of *Ducks, Four People, In to tree, Johnny, Kristen and Sara, Old town, Parkjoy, Parkrun, Shields* and *Stockholm*, Then, the *Mixed CIF* sequence, composed of *Akiyo, News, Coastguard, Foreman, Tennis, Soccer, Football, Stefan, Mobile* and *Husky*.

The process is performed GoP by GoP with GoP-size of $4, 8, 16, 32$ frames as in [15]. Each frame is classically split into 64 chunks [15], [17], [19], [33] or 256 chunks as in [25], [35]. Transmissions through AWGN channels in the CSNR range of $[0{\sim}30\text{dB}]$ are considered as in [15], [17], [25], [33]. Four CR=1, 0.75, 0.5 and 0.25 are considered.

Due to limited space, among the different configurations (video content, GoP-sizes, etc.), only the results for the HD720p sequences are shown in this paper. The results for the other sequences are similar and can be found in the following link: https://drive.google.com/drive/folders/1nHqEkz8uwjs9-Sw7BHRHtXTx2URbOmYx?usp=sharing. The GoP-size and the number of chunks per frame are respectively set to 16 frames and 64 chunks per frame as it represents the original and mostly used configuration [15]. We verified similar results for the other video sequences, GoP-sizes, and chunk-sizes.

Fig. 6 presents the comparison between our model and the full end-to-end transmission simulations of the SoftCast(ZF) scheme for different CR and CSNR values.

Results show that:

- When CR=1 (red color), we logically obtain the same linear characteristic as in [1];
- However, the model proposed in [1] is no longer valid when the available channel bandwidth decreases (cyan, green and blue curves) as the data compression is not considered. In practice, it is mandatory to consider such loss since it drastically degrades the received video quality and implies non-linear characteristics at high CSNR. This is the well-known leveling-off effect [6] that appears and implies huge decibel losses (e.g., ΔPSNR_1=21.5dB for the considered case). This phenomenon is perfectly described by our model. In terms of video quality, the PSNR difference between two curves denoted by ΔPSNR becomes lower as the CSNR decreases. For instance, the ΔPSNR between the CR=1 (red curves) and CR=0.25 (blue curves) cases decreases from 21.5dB to 3.5dB when the CSNR goes from 30dB to 0dB. This is perfectly explained with the proposed model where for low CSNR values ($<$ 10dB), losses due to noise takes precedence over losses due to compression.
- The reconstructed PSNR goes below 35dB when considering very low CSNR values. Note that for such low CSNR values, classical standards such as H.264/AVC or HEVC offer worse quality and suffer glitches due to severe decoding errors [45] as illustrated in Fig. 2b in the Introduction. On the contrary, SoftCast can deal with any channel quality by delivering, at low CSNR, a video signal with low but acceptable quality [15] (see Fig. 2c);
- Since there is no approximation in the derivation process of (27), our model perfectly matches the simulations for all the considered bandwidths and CSNR values. However, as mentioned in Section II, the LLSE estimator is more commonly used, as in the original works of Jakubczak *et al.* [15] or as reported in [16]–[20], [24]–[28], [30]–[37]. Therefore, we propose in the next section a new model that incorporates the benefits of using the LLSE estimator.

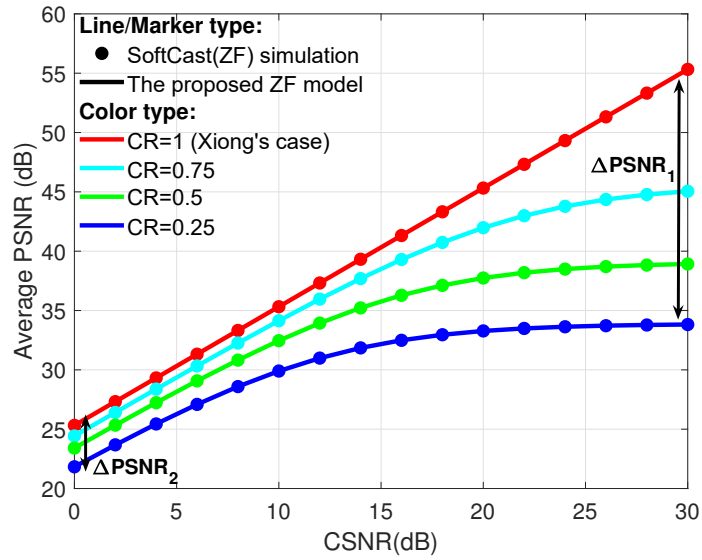


Fig. 6. Average PSNR results for the proposed theoretical model (solid lines) and SoftCast simulations with ZF estimator: (dots markers) for the *Mixed HD720p* sequence. Configuration: GoP-size=16 frames, 64 chunks/frame. Colors red, cyan, green and blue represent CR=1, 0.75, 0.5 and 0.25, respectively.

IV. Description of the proposed models considering LLSE estimator and near-optimal power allocation

Recall Fig. 4 and considering the LLSE estimator [15], [17] instead of the ZF one, (7) becomes:

$$\begin{aligned} \hat{x}_i &= \hat{y}_i \cdot \alpha_i, \\ &= \hat{y}_i \cdot \frac{g_i \lambda_i}{g_i^2 \lambda_i + \sigma_n^2}. \end{aligned} \tag{28}$$

Likewise, (8) is changed to:

$$\begin{aligned} D_{i[\text{LLSE/FB}]} &= E\big[(\hat{x}_i - x_i)^2\big], \\ &= \frac{\sigma_n^2 \alpha_i}{g_i}, \\ &= \frac{\sigma_n^2 \lambda_i}{g_i^2 \lambda_i + \sigma_n^2}, \\ &= \frac{\sigma_n^2 \lambda_i}{P_i + \sigma_n^2}. \end{aligned} \tag{29}$$

(6) and (9) remain valid for this case.

As above, we want to express P_i as a function of $D_{i[\text{LLSE/FB}]}$. For this purpose, let us recall the near-optimal power allocation defined by the following scaling factor equation (2) from [15] (see Section II): $g_i^2 = \frac{P}{\sqrt{\lambda_i}\sum_i \sqrt{\lambda_i}}$. Inserting this into (9), we get:

$$\begin{aligned} P_i &= g_i^2 \lambda_i, \\ &= \frac{P\sqrt{\lambda_i}}{\sum_i \sqrt{\lambda_i}}. \end{aligned} \tag{30}$$

Therefore, the distortion per element can be expressed as:

$$\begin{aligned} D_{i[\text{LLSE/FB}]} &= \frac{\sigma_n^2 \lambda_i}{\frac{P\sqrt{\lambda_i}}{\sum_i \sqrt{\lambda_i}} + \sigma_n^2}, \\ &= \frac{\lambda_i}{\frac{\sqrt{\lambda_i}}{\sum_i \sqrt{\lambda_i}} \cdot \frac{P}{\sigma_n^2} + 1}, \\ &= \frac{\lambda_i}{\frac{\sqrt{\lambda_i}}{\sum_i \sqrt{\lambda_i}} \cdot (\text{CSNR}_{lin} \cdot N) + 1}. \end{aligned} \tag{31}$$

By combining (17) and (30), the LLSE distortion equation can be also given according to the ZF one as follows:

$$D_{i[\text{LLSE/FB}]} = D_{i[\text{ZF/FB}]} \cdot \frac{1}{1 + \frac{\sigma_n^2}{P_i}}. \tag{32}$$

Assuming that the sender transmits all the N elements of $\mathbf{x}$ (*i.e.*, no channel bandwidth restriction) and that the LLSE estimator is used at the receiver side, the total expected distortion under near-optimal power allocation for the LLSE case, denoted by $\text{D}_{[\text{LLSE/FB}]}$ is given by:

$$\text{D}_{[\text{LLSE/FB}]} = \sum_{i=1}^{N} D_{i[\text{LLSE/FB}]}. \tag{33}$$

Recall (19), (20), (31) and (32), the expected reconstructed video quality considering the LLSE estimator without data compression can be obtained from:

$$\text{PSNR}_{[\text{LLSE/FB}]} = c - 10\log_{10}\left(\frac{1}{N}\sum_{i=1}^{N} \frac{\lambda_i}{\frac{\sqrt{\lambda_i}}{\sum_i \sqrt{\lambda_i}} \cdot (\text{CSNR}_{lin} \cdot N) + 1}\right). \tag{34}$$

(34) has a straightforward expression by considering the approximation $P_i \simeq P/N$, *i.e.*, a flat power allocation. In the following, equations resulting from this approximation are marked by an asterisk. Recall (32) and (33), $\text{D}_{[\text{LLSE/FB}]}$ becomes:

$$\text{D}_{[\text{LLSE/FB}]^*} = \sum_{i=1}^{N} D_{i[\text{ZF/FB}]} \cdot \frac{1}{1 + \frac{N\sigma_n^2}{P}}. \tag{35}$$

Recall (19) and (20), the expected reconstructed video quality considering the LLSE estimator without data compression is given by:

$$\begin{aligned} \text{PSNR}_{[\text{LLSE/FB}]^*} &= c - 10\log_{10}\left(\frac{\text{D}_{[\text{LLSE/FB}]^*}}{N}\right), \\ &= c - 10\log_{10}\left(\text{D}_{[\text{ZF/FB}]}\right) \\ &\quad + 10\log_{10}\left(1 + \frac{1}{\text{CSNR}_{lin}}\right), \\ &= c + \text{CSNR} - 20\log_{10}\left(H\right) \\ &\quad + 10\log_{10}\left(1 + \frac{1}{\text{CSNR}_{lin}}\right). \end{aligned} \tag{36}$$

When the transmitted signal follows a flat power distribution, and only in this case, the theoretical model of the SoftCast-based scheme assuming near-optimal power allocation and LLSE estimator is given by:

$$\text{PSNR}_{[\text{LLSE/FB}]^*} = \text{PSNR}_{[\text{ZF/FB}]} + G_{\text{LLSE}}, \tag{37}$$

where $G_{\text{LLSE}} = 10\log_{10}\left(1 + \frac{1}{\text{CSNR}_{lin}}\right)$.

As seen in section III-B, this model can be extended to the CB case by adding an additional term that represents the discarded elements:

$$\begin{aligned} \text{PSNR}_{[\text{LLSE/CB}]^*} &= c + \text{CSNR} - 20\log_{10}\left(H_t\right) \\ &\quad + G_{\text{LLSE}} \\ &\quad - 10\log_{10}\left(1 + \frac{(\text{CSNR}_{lin} + 1) \cdot E_d}{{H_t}^2}\right), \\ &\simeq \text{PSNR}_{[\text{ZF/CB}]} + G_{\text{LLSE}}. \end{aligned} \tag{38}$$

Proof is available in Appendix A.

In a similar way, the expected reconstructed video quality considering the general case ($P_i \not\approx P/N$), the LLSE estimator and CB applications can be obtained from:

$$\text{PSNR}_{[\text{LLSE/CB}]} = c - 10\log_{10}\left({D_s}/{N} + {D_d}/{N}\right), \tag{39}$$

where D_s and D_d are obtained in a similar way to (23).

Note that (38) is similar to (27), except that the fifth and last term includes $(\text{CSNR}_{lin} + 1)$ instead of (CSNR_{lin}), due to the use of the LLSE estimator. From (37) and (38), it can be seen that the maximum difference between the ZF model and LLSE$*$ model corresponds to the G_{LLSE} term. We give in Table I the corresponding numerical values of G_{LLSE} for several CSNR values.

TABLE I
EVALUATION OF THE PSNR IMPROVEMENT BROUGHT BY THE LLSE ESTIMATOR IN COMPARISON TO THE ZF ESTIMATOR

CSNR(dB)	0	5	10	15	20	25
G_{LLSE}(dB)	3.01	1.19	0.41	0.13	0.04	0.01

We can conclude from Table I that above 10dB, the improvements brought by the LLSE estimator in terms of PSNR scores are insignificant. This is consistent with [1], [15] and it is confirmed below.

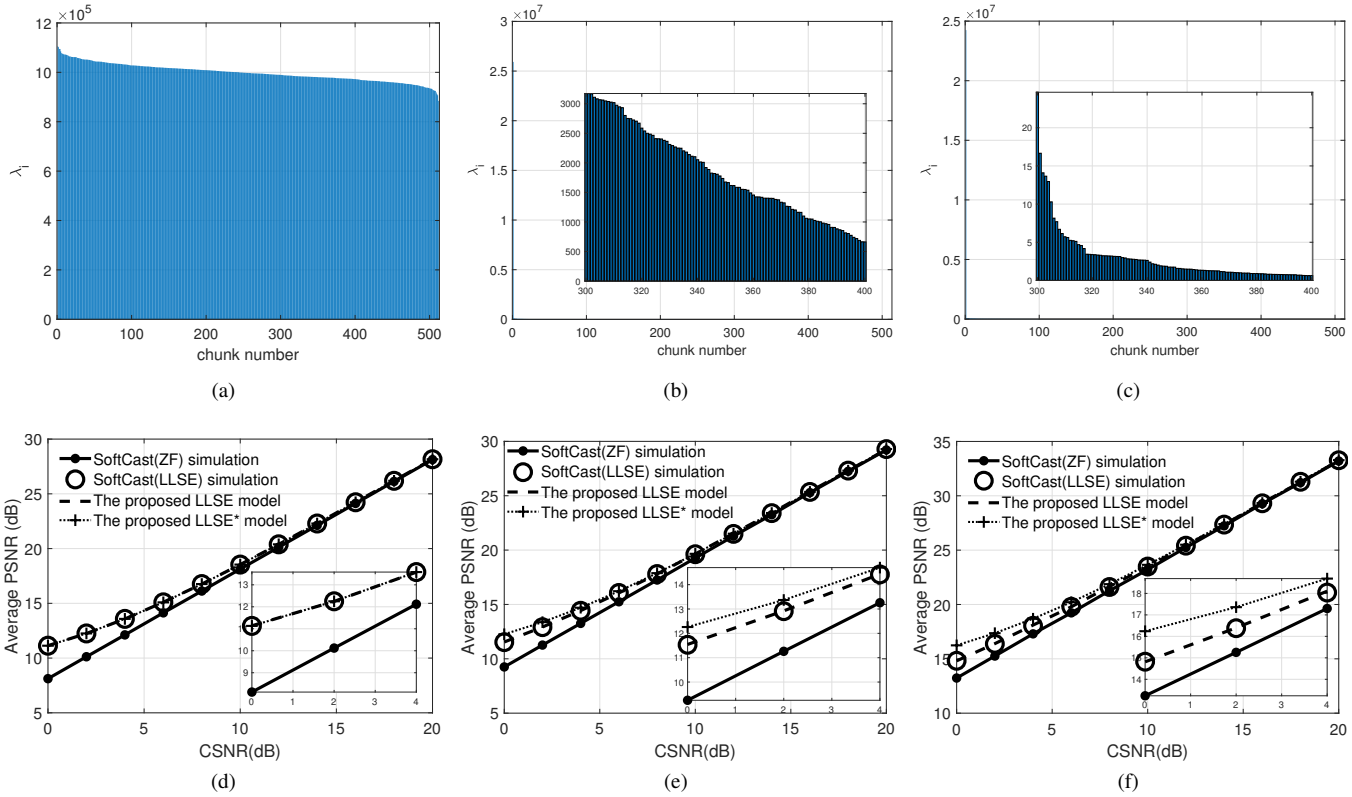


Fig. 7. Average PSNR results for the proposed LLSE theoretical model (dashed lines), the LLSE simplified model (dotted lines with cross markers), the SoftCast(LLSE) simulations: (big circle markers) and SoftCast(ZF) simulations: (solid lines with dot markers) for randomly generated power allocations. Configuration: GoP-size=8 frames, 64 chunks/frame, CR=1. First row, illustration of randomly generated power allocations. Second row: Corresponding Average PSNR results. (a), (d) Amplitude=[1000*randn(1,512)], (b), (e) Amplitude=[5000 100*randn(1,511)], (c), (f) Amplitude=[5000 100*randn(1,311) randn(1,200)]. Please enlarge the figure to observe details. The small figures in (b), (c), (d), (e), (f) correspond to a zoom of the main figures.

In the following, we verify the validity of the simplified model through simulations. We recall that the simplified model is based on the following assumption $P_i \simeq P/N$ (or $P_i \simeq P/K$ in case of bandwidth constraints), *i.e.*, the signal follows a flat power distribution. In practice, this distribution is unlikely to happen, it would require a video sequence with uncorrelated data, or in other words a video with very high spatio-temporal activity/complexity. One way to obtain such distribution is to create images/video based on randomly generated uncorrelated (DCT) coefficients. First, we generate a Random Power Distribution (RPD) representing a quasi-flat power distribution. Then, in order to evaluate the PSNR gap between the simplified model and simulations we create two others RPD representing more realistic power distributions that may be observed with real video. Finally, the PSNR gap is also evaluated using real power distributions from still images/video content.

To create the RPD, we use normally distributed random numbers, playing the role of chunks, as the input of the scheme. Precisely, a matrix of 512 chunks composed each of 36×44 random coefficients is generated as typically done in a SoftCast-based scheme, assuming CIF sequences and GoP-size of 8 frames. Each chunk is then multiplied by a gain defined hereafter to modify the shape of the power distribution. We assume without loss of generality that all the chunks are transmitted (*i.e.*, CR=1).

Transmissions through AWGN channels in the range of $[0{\sim}20\text{dB}]$ are considered as the effect of the LLSE estimator is limited for higher CSNR.

The first RPD representing the quasi-flat power distribution (no major difference of power between each chunk) is illustrated in Fig. 7a. Simulation results are given in Fig. 7d. Since the approximation of a flat power distribution is respected, the limit represented by (36) in dotted lines with cross markers can be reached by the SoftCast(LLSE) scheme. This is verified both by simulations (big circle markers) and theory through (34) represented by dashed lines.

Then, we consider the second and third RPD that simulate typical power distribution observed with real video. They both contain one high value (that represents the chunk containing the low frequencies including the DC component, thus of high energy) and 511 others. The second RPD contains relative high energy for these 511 values as illustrated in Fig. 7b. It represents a high spatiotemporal complexity video signal (e.g., such as the *Husky* video sequence). The third RPD shown in Fig. 7c represents a low complexity video signal (e.g., such as the *Akiyo* video sequence) where 200 values are left almost nulls. Results for these two power distributions are available in Fig. 7e and Fig. 7f, respectively. We observe that:

- The SoftCast(LLSE) simulations and model are delimited by the simplified LLSE model (LLSE*) and by the SoftCast(ZF) simulations/model;

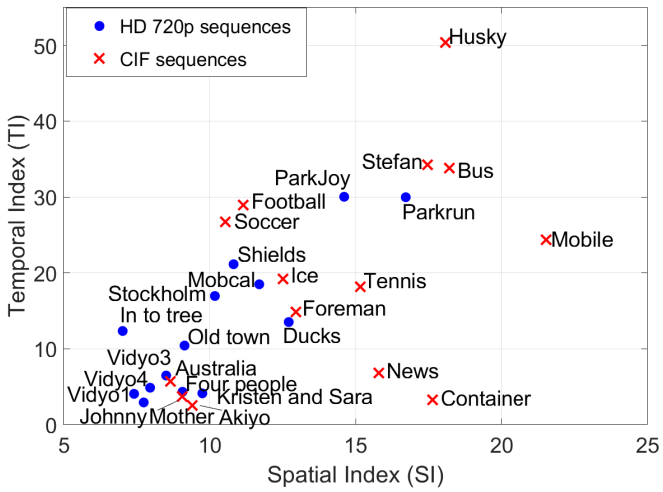


Fig. 8. Illustration of the average spatiotemporal indexes for the selected HD and CIF video sequences.

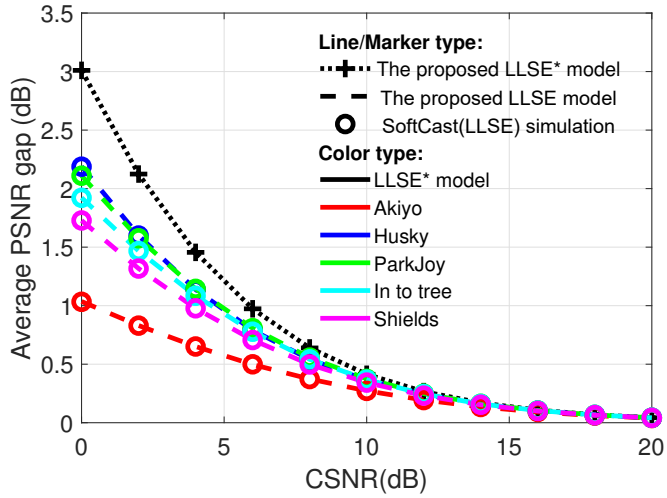


Fig. 9. Illustration of the PSNR gap between the LLSE estimator and ZF estimator. Configuration: GoP-size = 16 frames, *full* available bandwidth. Colors: black = LLSE model*, red = *Akiyo*, blue = *Husky*, green = *ParkJoy*, cyan = *In to tree* and magenta = *Shields* video sequences.

- As the power distribution becomes heterogeneous, the gain brought by the LLSE estimator decreases and thus, the PSNR gap between the two proposed models (*i.e.*, LLSE and LLSE*) becomes larger.

We finally examine the impact of real video content on the existing PSNR gap between the ZF estimator and LLSE estimator. To characterize the video content, we use the amount of Spatial and Temporal information in a video sequence, defined by the (SI) and (TI) indexes proposed by the ITU-T [46]. They are defined as follows

$$\text{SI} = max_{time}\{std_{space}[Sobel(I(i,j,k))]\}, \quad (40)$$
$$\text{TI} = max_{time}\{std_{space}[I(i,j,k) - I(i,j,k-1)]\}, \quad (41)$$

where $I(i,j,k)$ represents the kth frame, (i,j) the corresponding spatial coordinates and $Sobel()$ the Sobel filtering operation, respectively.

However, as mentioned in [47] due to the current definition of these indexes that select the highest value along the time axis, performing the TI computation for a video with slow motions that contains cuts results in a high TI value. In order to have more representative (SI, TI) values, we choose to average the results over the entire sequence. The new definitions are

$$\text{SI} = mean_{time}\{std_{space}[Sobel(I(i,j,k))]\}, \quad (42)$$
$$\text{TI} = mean_{time}\{std_{space}[I(i,j,k) - I(i,j,k-1)]\}. \quad (43)$$

These definitions are considered in the rest of this paper instead of eq. (40) and eq. (41). Fig. 8 shows the resulting average (SI, TI) values for each sequence.

The PSNR gap between the LLSE estimator and ZF estimator has been quantified for all the HD and CIF sequences in Fig. 8. A representative summary of the results with five selected sequences is shown in Fig. 9. We verified similar behaviors for the other video content. Results show that:

- As mentioned above, the PSNR gap between the ZF and LLSE estimator varies according to the video characteristics or more precisely, the power distribution of the transmitted chunks/elements;
- This gap is always under the LLSE* model that seems to define a maximum limit that can be reached when the signal is uncorrelated and spread over the chunks, *i.e.*, when the power is equally distributed;
- As the spatiotemporal indexes increase, the performance of the LLSE estimator increases as observed with the *Husky* or *ParkJoy* sequences, *i.e.*, when the video signal is difficult to decorrelate due to high motions and strong edges. On the contrary, the performance of the LLSE estimator decreases when the spatiotemporal indexes are small. Indeed, in such cases, the correlation is high and thus, after the decorrelation transform, most of the energy is located on the low frequency chunks leaving only fragments of energy for the others chunks, as similarly observed in Fig. 7c and Fig. 7f with the RPD signals.

Based on these observations and for a quick evaluation of the LLSE's performance, we suggest to use the LLSE simplified model, *i.e.*, (38) for high spatiotemporal content or (27) for low spatiotemporal content as the bias is limited. For a precise evaluation, and specifically at low CSNR values, we recommend using the un-simplified theoretical model (34) for FB case and (39) for CB cases as they do not introduce bias between them and the end-to-end simulations, in contrast to the simplified ones.

As before, we also give the comparison between our models and the full end-to-end transmission simulations for SoftCast(LLSE) considering the *Mixed HD720p* sequence in Fig. 10. As observed, the simulation results lie in between the ZF and LLSE* model since the composite sequence contains 10 different sequences. As the LLSE model is not approximated, the predicted values offered by the latter and the values obtained by simulations are equal.

We have seen in this section the theoretical model of the SoftCast-based scheme assuming LLSE estimator at the receiver side and near-optimal power allocation at the receiver, *i.e.*, power allocation considering no channel feedback at the transmitter. This constitutes the main scheme in the literature,

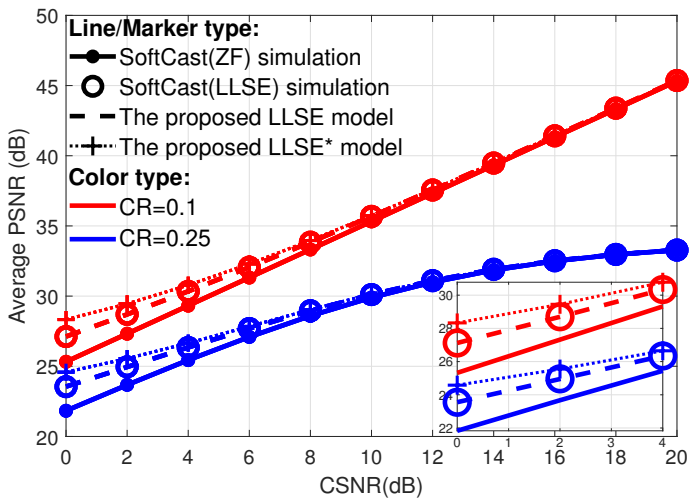


Fig. 10. Average PSNR results for the proposed LLSE theoretical models: (LLSE model: dashed lines, LLSE* model: dotted lines and cross markers) and SoftCast simulations: (LLSE: big circle markers, ZF: solid lines with dot markers) for the *Mixed HD720p* sequence. Configurations: GoP-size=16 frames, 64 chunks/frame. Colors red and blue represent CR=1 and CR=0.25, respectively.

as it is suitable for broadcast context, where only one data stream is sent and can be decodable by any receiver. However, one may also find, SoftCast+ scheme [31], [32]. We give in the next section the corresponding theoretical model.

V. Description of the proposed model considering LLSE estimator and Optimal Power Allocation

In this section, we evaluate the performance of SoftCast+ [32] *i.e.*, where the optimal power allocation is performed and the LLSE estimator is used at the receiver side. In this case, the $N-\ell$ chunks/elements with power level below the noise power level (σ_n^2) are discarded, and the total available power P is reassigned to the ℓ transmitted chunks/elements.

Therefore, as for the ZF/CB case, the total distortion consists of two terms: $\mathrm{D}_{[\text{OPA-LLSE}]} = D_s + D_d$, where D_s and D_d correspond respectively to the total distortion due to the ℓ transmitted elements and the distortion due to the $N-\ell$ discarded elements with energy below noise power level. The way to compute ℓ, the optimal number of transmitted elements taking into account the CSNR value is detailed later on.

The expected distortion for each of the transmitted element $\hat{x}_i$ is the same since LLSE estimator is used at the receiver side (see section IV):

$$D_i = E\left[(\hat{x}_i - x_i)^2\right] = \frac{\sigma_n^2 \lambda_i}{P_i + \sigma_n^2}. \tag{44}$$

The new optimization problem is defined as follows:

$$(P2): \min \sum_{i=1}^{\ell} D_i + \sum_{i=\ell+1}^{N} D_j, \text{s.t.} \sum_{i=1}^{\ell} P_i \leq P \tag{45}$$

This is a Lagrangian problem:

$$\mathcal{L} = \sum_{i=1}^{N} D_i + \frac{1}{C^2_{\text{OPA-LLSE}}} \sum_{i=1}^{N} P_i, \tag{46}$$

where $C^2_{\text{OPA-LLSE}}$ is the Lagrange multiplier.

Differentiating (46) with respect to P_i and setting the result to zero, we get:

$$\sigma_n^2 \frac{\lambda_i}{(P_i + \sigma_n^2)^2} = \frac{1}{C^2_{\text{OPA-LLSE}}}. \tag{47}$$

This determines the optimal power for sending x_i:

$$P_i = C_{\text{OPA-LLSE}} \sigma_n \sqrt{\lambda_i} - \sigma_n^2. \tag{48}$$

Note that only the elements with $P_i > 0$ are transmitted. From (48) we get the following condition:

$$\lambda_i > \frac{\sigma_n^2}{C^2_{\text{OPA-LLSE}}}. \tag{49}$$

Since there exists a constraint on the total power transmission, $C^2_{\text{OPA-LLSE}}$ can be further defined. Indeed, without loss of generality, let us assume that the P_i $(i = 1, 2, \ldots, N)$ are ordered in descending order and that only the ℓ elements verify the preceding inequality, then:

$$\begin{aligned} \sum_{i=1}^{\ell} P_i &= P, \\ \sum_{i=1}^{\ell} C_{\text{OPA-LLSE}} \sigma_n \sqrt{\lambda_i} - \sigma_n^2 &= P, \\ C_{\text{OPA-LLSE}} \sigma_n \sum_{i=1}^{\ell} \sqrt{\lambda_i} - \ell \sigma_n^2 &= P, \\ C_{\text{OPA-LLSE}} \sigma_n \sum_{i=1}^{\ell} \sqrt{\lambda_i} &= P + \ell \sigma_n^2. \end{aligned} \tag{50}$$

Finally,

$$C_{\text{OPA-LLSE}} = \frac{P + \ell \sigma_n^2}{\sigma_n \sum_{i=1}^{\ell} \sqrt{\lambda_i}}. \tag{51}$$

Recall the equations (44) and (48), we easily get:

$$\begin{aligned} D_i &= \frac{\sigma_n^2}{P_i + \sigma_n^2} \lambda_i, \\ &= \frac{\sigma_n^2}{C_{\text{OPA-LLSE}} \sigma_n \sqrt{\lambda_i}} \lambda_i, \\ &= \frac{\sigma_n}{C_{\text{OPA-LLSE}}} \sqrt{\lambda_i}. \end{aligned} \tag{52}$$

Finally, the total distortion for the LLSE case with optimal power allocation is given by [31], [32]:

$$\begin{aligned} \mathrm{D}_{[\text{OPA-LLSE}]} &= \sum_{i=1}^{\ell} D_i + \sum_{j=\ell+1}^{N} D_j, \\ &= \frac{\sigma_n^2 \left(\sum_{i=1}^{\ell} \sqrt{\lambda_i}\right)^2}{P + \ell \sigma_n^2} + \sum_{j=\ell+1}^{N} \lambda_j. \end{aligned} \tag{53}$$

We note that the definition of $\mathrm{D}_{[\text{OPA-LLSE}]}$ (53) is close to $\mathrm{D}_{[\text{ZF/CB}]}$ (23) except that:

1) the number of discarded chunks is ℓ and it varies according to the channel noise power,

$$\text{PSNR}_{[\text{OPA-LLSE}]} = c + \text{CSNR} + G_{\text{LLSE}} - 20\log_{10}\left(H_{t2}\right) - 10\log_{10}\left(1 + \frac{(\text{CSNR}_{lin}+1)\cdot E_{d2}}{{H_{t2}}^2}\right). \quad (57)$$

2) there is an additional term $(\ell\sigma_n^2)$ at the denominator that represents the effect of the LLSE estimator.

Therefore, the development is similar to section III-B, and we get:

$$\begin{aligned}\text{PSNR}_{[\text{OPA-LLSE}]} &= 10\log_{10}\left(\frac{255^2}{{D_s}/{N} + {D_d}/{N}}\right), \quad (54)\\ &= c - 10\log_{10}\left(1+\frac{D_d}{D_s}\right)\\ &\quad + 10\log_{10}\left(\frac{\bar{P}+\sigma_n^2}{\sigma_n^2}\right)\\ &\quad - 10\log_{10}\left(\frac{1}{N\ell}\left(\sum_{i=1}^{\ell}\sqrt{\lambda_i}\right)^2\right).\end{aligned}$$

Please note that the average transmission power in (20) becomes $\bar{P} = P/\ell$ as the total transmission power is here distributed over the ℓ transmitted elements of $\mathbf{x}$.

By analogy with (24), we identify the new data activity of the remaining transmitted elements as:

$$H_{t2} = \frac{1}{\sqrt{N\ell}}\sum_{i=1}^{\ell}\sqrt{\lambda_i}. \quad (55)$$

For the ease of reading we also define E_{d2}, the overall energy of all discarded elements:

$$E_{d2} = \frac{1}{N}\sum_{j=\ell+1}^{N}\lambda_j. \quad (56)$$

According to these new definitions, the end-to-end video quality for the LLSE estimator with optimal power allocation is finally given by (57) at the top of the page. Proof can be found in Appendix B.

The equation (57) has a similar form as (27). Except that:

- Like (38), it includes the G_{LLSE} term that reflects the benefits of the LLSE estimator. However, unlike (38), this new model is valid regardless of the power distribution;
- The fifth and last term includes $(\text{CSNR}_{lin}+1)$ like (38), instead of (CSNR_{lin}) for (27);
- The definition of H_{t2} and E_{d2} depends on ℓ instead of K.

Whereas in (38), the elements/chunks are only discarded due to bandwidth constraints, in (57), in order to optimize the received quality, SoftCast+ may discard some elements/chunks even if the bandwidth available at the transmitter allows transmitting all of them. As a consequence, the above model already includes both FB and CB cases. For the constrained-bandwidth applications, ℓ actually represents the minimum value between the number of discarded elements Nb_1 due to optimal power allocation and the number of discarded elements Nb_2 that match the available bandwidth ($\ell = min(Nb_1, Nb_2)$). We note that the number of discarded elements for optimal power allocation is not fixed and depends on the channel characteristics. It is updated for each received CSNR estimate at the transmitter.

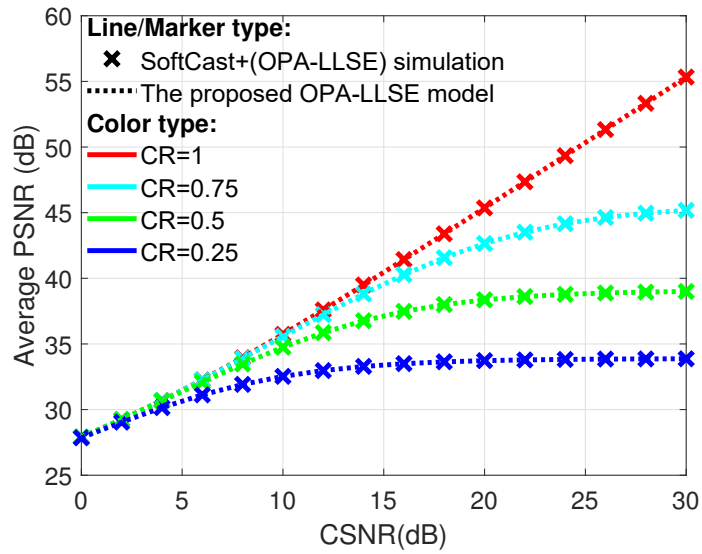


Fig. 11. Average PSNR results for the proposed theoretical model (dotted lines) and SoftCast+ simulations (optimal power allocation with LLSE estimator): (cross markers) for the *Mixed HD720p* sequence. Configurations: GoP-size=16 frames, 64 chunks/frame. Colors red, cyan, green and blue represent CR=1, 0.75, 0.5 and 0.25, respectively.

Like the other models, assuming that the value of E_{d2} is transmitted as a unique additional metadata, it is possible for the receiver to compute and estimate the PSNR score of the reconstructed video even without having the original one.

The effectiveness of the proposed model is compared to the SoftCast+ scheme (using optimal power allocation and LLSE estimator). We use the same simulation configurations given in Section III-B. Results given in Fig. 11 show that:

- In contrast to previous models, where bandwidth constraints directly imply a loss of quality even at low CSNR values, SoftCast+ gives almost the same received quality for both cases at low CSNR (e.g., CSNR$\leq$ 5dB for this video sequence). This is because for such low CSNR values, the number of transmitted chunks with SoftCast+ is usually really small (e.g., only $\frac{206}{1024}$ and $\frac{276}{1024}$ chunks per GoP in average respectively for a CR=0.25 and a CR=1 for the *Mixed HD720p* sequence under a CSNR=0dB, meaning that only $\sim$ 20% of the total bandwidth is used).
- In all cases, our model perfectly matches the simulations over the entire CSNR range, independently of the bandwidth case considered. This is predictable since no approximation is made in the derivation process of (57).

VI. Global Performance Evaluation of the Models

In this section, we compare the performance of the three SoftCast schemes through our models. In addition, we give an example of the possible use of these models. Specifically, by

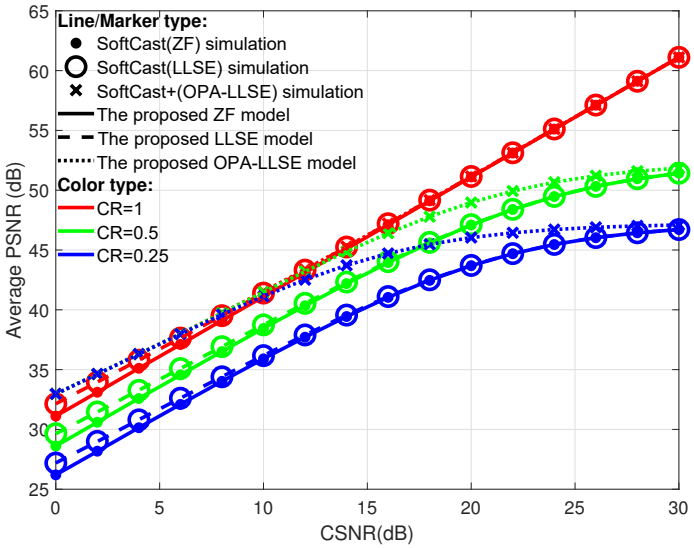


Fig. 12. Average PSNR results for the proposed theoretical models: ZF (solid lines), LLSE (dashed lines) and OPA-LLSE (dotted line); and SoftCast simulations: SoftCast ZF (dots), SoftCast LLSE (circle markers) and SoftCast+ OPA-LLSE (cross markers) for the *Johnny* sequence. Configuration: GoP-size=16 frames, 64 chunks/frame. Colors: red, green and blue represent CR=1, 0.5 and 0.25, respectively.

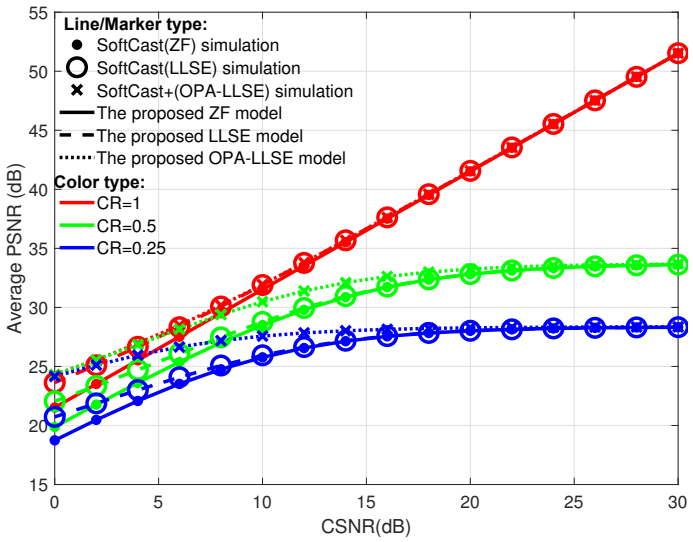


Fig. 13. Average PSNR results for the proposed theoretical models: ZF (solid lines), LLSE (dashed lines) and OPA-LLSE (dotted lines); and SoftCast simulations: SoftCast ZF (dots), SoftCast LLSE (circle markers) and SoftCast+ OPA-LLSE (cross markers) for the *ParkJoy* sequence. Configuration: GoP-size=16 frames, 64 chunks/frame. Colors: red, green and blue represent CR=1, 0.5 and 0.25, respectively.

using them, we theoretically validate the experimental results from our previous work [33].

A. Comparison of the three schemes

We first compare the end-to-end performance of the proposed models and their corresponding SoftCast schemes: SoftCast(ZF), SoftCast(LLSE) and SoftCast+.

The parameters used in the simulations are the same as described in Section III-B. Among all the video sequences, we choose to show the results for the *Johnny* and *ParkJoy* video sequences because of their spatiotemporal information disparities as observed in Fig. 8. For clarity, we only show the results for CR=1, 0.5 and 0.25 represented by red, green and blue colors, respectively. Results for the other CR values are similar.

Results in Fig. 12 and Fig. 13 show that:

- For low CSNR values $\leq$ 10dB, as already known and regardless of the transmitted video content, the LLSE estimator (dashed lines and circle markers) outperforms the ZF one (solid lines and dots markers). However, the PSNR improvement is limited and at a maximum still low as shown in Table I in Section IV;
- Regardless of the configuration (GoP-size, available bandwidth, transmitted video content), SoftCast(ZF) offers the worst PSNR, followed by SoftCast(LLSE) which performs worse than SoftCast+. Note that SoftCast+ employs both LLSE estimator and optimal power allocation;
- When comparing the SoftCast(LLSE) (dashed lines and circle markers) to the SoftCast+ (dotted lines and cross markers), we can observe that SoftCast+ offers a marginal performance improvement as stated in [34]. However, this is only true when considering no bandwidth restriction (CR=1). For instance, let us consider a CR=0.25 and a CSNR=0dB, the PSNR gap between these two versions is about 5.76dB and 3.43dB, respectively for the *Johnny* and *ParkJoy* video sequences. This gap decreases as the CSNR increases and becomes almost null after a CSNR$\geq$30dB (resp. CSNR$\geq$15dB), for the *Johnny* (resp. *ParkJoy*) video sequence. This is perfectly explained by the proposed models, where for the *ParkJoy* sequence, most of the chunks should be transmitted but cannot due to the bandwidth constraints. In contrast, for the *Johnny* video sequence, the improvement over the classical SoftCast(LLSE) is still important even above CSNR=15dB, due to the fact that most of the chunks are energy-limited and can be discarded to smartly reallocate the total power available at the transmitter.

Based on the power distribution (λ_i) directly obtained at the transmitter after the decorrelation transform, one can quickly evaluate the performance of the considered scheme without having to perform full extensive end-to-end simulations.

In addition, we use our models to show that the performance and behaviors of SoftCast-based schemes are highly dependent on the transmitted content. In [33], we demonstrated that the GoP-size is of paramount importance in a SoftCast context. Specifically, depending on the spatiotemporal characteristics of the video and the intended application, we showed that an optimal GoP-size could be defined either to improve the received quality or to decrease the complexity while offering similar PSNR scores. In the following, the proposed models are used to predict the optimal GoP-size for the three SoftCast-based schemes considering different video content.

B. Example of application

We take into account different GoP-size=4, 8, 16 and 32 frames. Due to limited spaces, we only show the results

for CR=1 (first row), and CR=0.25 (second row). When compression is needed due to bandwidth constraints, we ensure to keep the same symbol rate for all the methods. For instance, for a CR=0.25, we keep the equivalent of 2 frames (resp. 4) for the GoP-size=8 (resp. 16). We verified that the results for other CR have similar behaviors. In this paper, the selected maximal GoP-size is set to 32 frames. Indeed, the complexity increases according to $O(Llog(L))$ with L the number of frames in a GoP [15], [48]. Choosing $L > 32$ implies high hardware capacities as well as an increase of the decoding time since the receiver needs to wait the L frames before processing the inverse temporal DCT. These two constraints may not be compatible with several practical applications.

The first selected video sequence is *Johnny*, which contains low spatiotemporal information. Regardless of the considered scheme and available channel bandwidth, results given in Fig. 14 show that increasing the GoP-size leads to a better received quality over the entire CSNR range. The PSNR gain between a GoP-size of 4 and 32 frames is about 6dB, regardless of the SoftCast scheme considered. This huge gain is due to the better use of the temporal DCT. Indeed, due to high temporal correlation (slow motions), using a larger GoP allows to better compact the information and hence reduces the *data activity*. However, and as explained before, when considering CB applications and high CSNR values ($\geq$25dB), the leveling-off effect appears. Therefore, the gain goes down and increasing the GoP-size does not bring large improvement.

Simulation results considering a high spatiotemporal content such as the *ParkJoy* video sequence are given in Fig. 15. In contrast to the *Johnny* sequence, the improvement is limited and increasing the GoP-size from 4 to 32 frames only brings about 1.2dB gain. Regardless of the studied scheme, the improvement is only about 0.2dB from a GoP-size equals 16 to 32 frames. Such improvement is insignificant as the MPEG committee considers that above 0.5dB, a difference is visually noticeable [41]. Therefore, we recommend using an intermediate GoP-size (8$\sim$16 frames) for such content. This is even more true as the gain between these two GoP-sizes quickly decreases and becomes null or slightly negative when considering CB applications and CSNR $\geq$15dB.

We verified that the two statements above are valid in average for all video content in Fig. 8. However, giving an optimal GoP-size considering only the characteristics of the video content itself is not easy since several other parameters such as the channel quality or the available bandwidth impact the received quality. In the following, we give general trends but recommend for a specific application, the use of the proposed models to quickly find the optimal GoP-size according to the available channel bandwidth and if known at the transmitter, the channel quality. If the delay induced by the decoding is not an important criterion, we suggest to use larger GoP-size (e.g., 32 frames) for low spatiotemporal information such as the *Johnny* or the *Akiyo* since under the same considered channel it brings larger gain in received video quality. On the other hand, using intermediate to small GoP-size (e.g., 8$\sim$16 frames) for high spatiotemporal content (sport events, etc.) such as the *ParkJoy* or the *Stefan* video sequences is sufficient since the gain between them and a larger GoP-size is insignificant or even negative. This is especially true for the *Husky* video sequence (high SI, very high TI) where increasing the GoP-size from 8 to 32 frames only brings less than 0.4dB improvement in average, which is unnoticeable [41]. Regardless of the video content, we note that the GoP-size of 4 frames is never preferred as it does not take enough advantage of the temporal correlation between frames. However, GoP-size of 4 frames or even smaller may be used for low-latency applications. In this case, our models can be used to quickly evaluate the PSNR loss induced by such GoP-size reduction.

We showed in this paper three models that can be used to predict the end-to-end performance of SoftCast-based schemes including:

- bandwidth-constrained applications
- LLSE estimator at the receiver side
- optimal power allocation

Depending on the targeted applications, we recommend using either (27) for broadcast context where one data stream is sent for all the receivers, or (57) for unicast context with CSNR channel quality feedback where optimal power allocation can be performed to improve the received quality. Indeed, (34) only brings a small improvement over (27) only at low CSNR values ($\leq$ 10dB). This improvement is limited as shown in Table I.

VII. Conclusion

In this paper, we provide a complete and comprehensive theoretical evaluation of the end-to-end performance of SoftCast-based linear video delivery schemes. Three theoretical PSNR-based models, which extend the model initially proposed in [1], are proposed by considering several realistic transmission scenarii. These models include 1) bandwidth-constrained applications 2) LLSE estimator 3) optimal power allocation. Predictions based on the models perfectly match the simulation results, hence accurately represent the full end-to-end SoftCast performance. These models can be used by the research community to evaluate their own algorithms against the SoftCast scheme without requiring full end-to-end extensive simulations. In contrast to the model proposed in [1], ours can help for parameter optimization in practical constrained-bandwidth applications where the linear relationship between video quality and CSNR is no longer valid over the entire CSNR range due to the appearance of the leveling-off [6] effect. Our models help to clearly characterize the origin of this phenomenon as well as to quantify the quality improvement brought by the LLSE estimator. Furthermore, to underline the utility of these models, we show that they can also be used to tune the parameters in a LVD context. For instance, we give general trends regarding an optimal GoP-size that can be selected according to the transmitted video content, *i.e.*, using a large GoP-size when transmitting low spatiotemporal complexity video content helps to increase the received quality. In contrast, increasing the GoP-size for high spatiotemporal complexity video content does not bring significant improvements. Finally, the models can also be used in real conditions to directly estimate the PSNR scores at the receiver side even without having the original video.

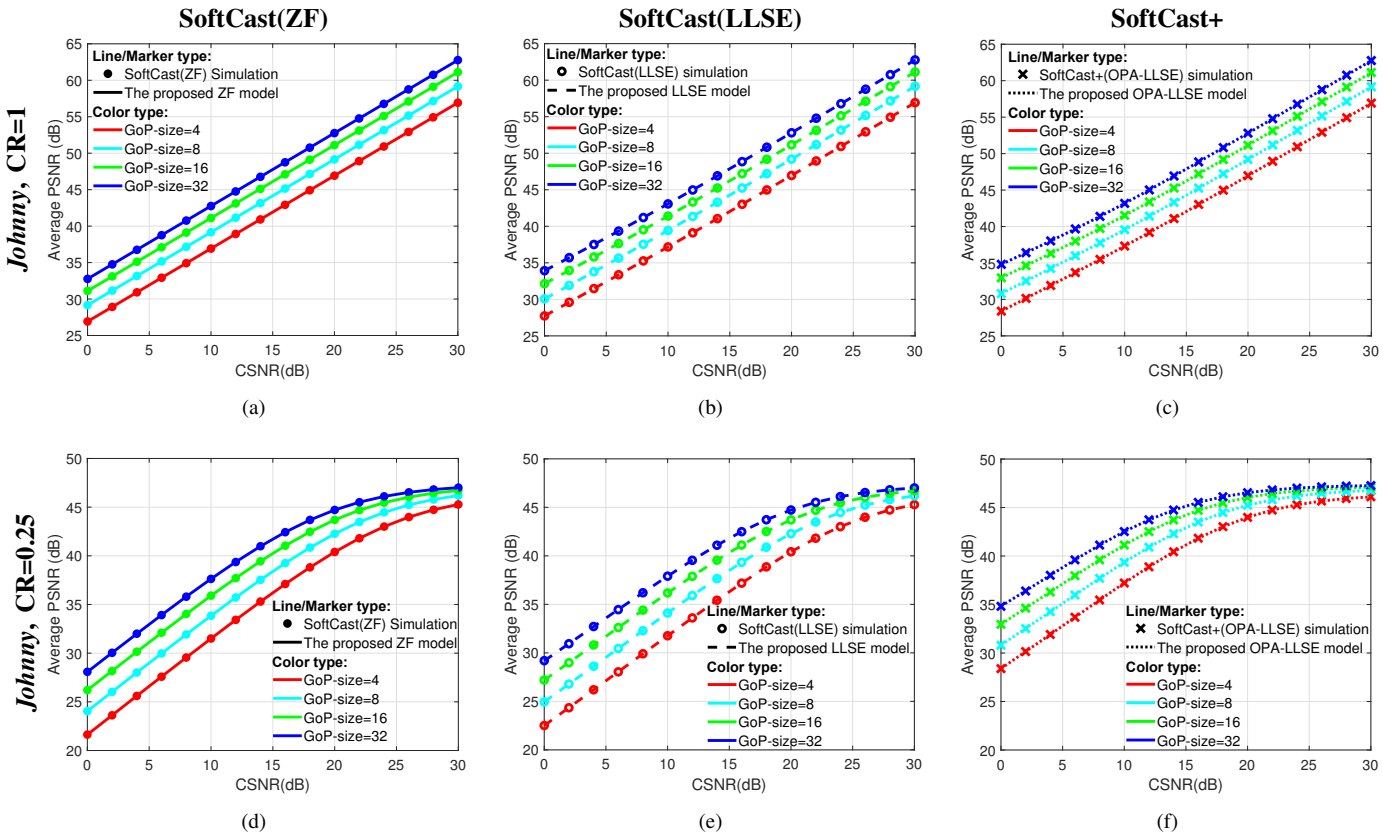


Fig. 14. Average PSNR results for theoretical models and SoftCast-based schemes for the *Johnny* sequence. (a), (b), (c): CR = 1. (d), (e), (f): CR=0.25. (a), (d): SoftCast(ZF) and theoretical ZF model. (b), (e): SoftCast(LLSE) and theoretical LLSE model. (c), (f): SoftCast+ and theoretical OPA-LLSE model.

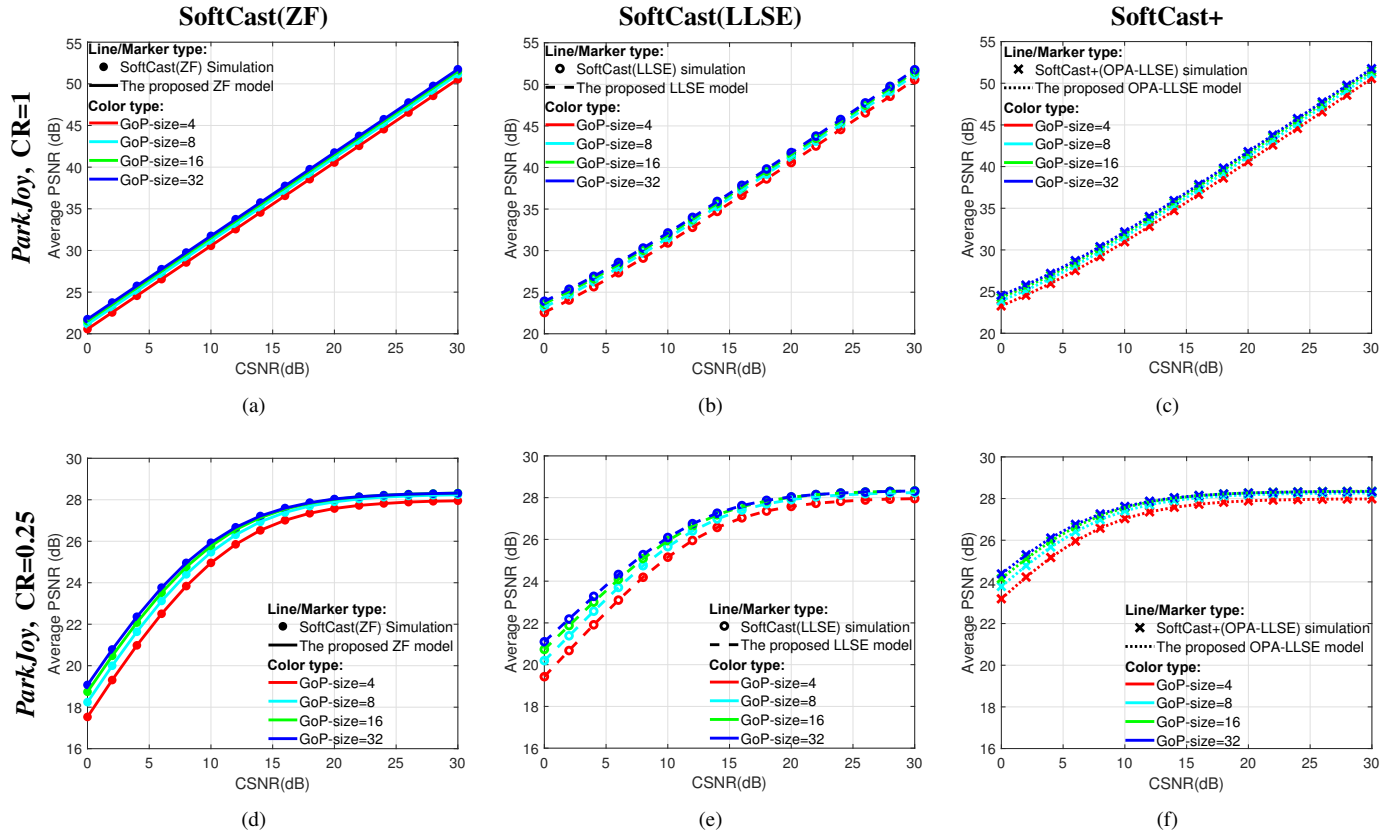


Fig. 15. Average PSNR results for theoretical models and SoftCast-based schemes for the *ParkJoy* sequence. (a), (b), (c): CR = 1. (d), (e), (f): CR=0.25. (a), (d): SoftCast(ZF) and theoretical ZF model. (b), (e): SoftCast(LLSE) and theoretical LLSE model. (c), (f): SoftCast+ and theoretical OPA-LLSE model.

APPENDIX A
PROOF OF THE EQUATION OF THE SIMPLIFIED LLSE MODEL FOR CB CASES

Proof. Considering the constrained-bandwidth cases, the total distortion for the simplified SoftCast(LLSE) model, denoted by LLSE* is defined by:

$$\begin{aligned} \mathrm{D}_{\text{[LLSE/CB]*}} &= \sum_{i=1}^{K} D_{i\text{[ZF/CB]}} \cdot \frac{1}{1+\frac{K\sigma_n^2}{P}} \\ &+ \sum_{j=K+1}^{N} \lambda_j. \end{aligned} \tag{A.1}$$

Recalling the equation of the CSNR (19) and PSNR (20), one gets:

$$\begin{aligned} \mathrm{D}_{\text{[LLSE/CB]*}} &= \sum_{i=1}^{K} D_{i\text{[ZF/CB]}} \cdot \frac{1}{1+\frac{1}{\mathrm{CSNR}_{lin}}} \\ &+ \sum_{j=K+1}^{N} \lambda_j. \end{aligned} \tag{A.2}$$

Using the property: $\log_{10}(a+b) = \log_{10}(a) + \log_{10}(1+\frac{b}{a})$, where $a = \sum_{i=1}^{K} D_{i\text{[ZF/CB]}} \cdot \frac{1}{1+\frac{1}{\mathrm{CSNR}_{lin}}}$ and $b = \sum_{j=K+1}^{N} \lambda_j$, and recalling that $D_{i\text{[ZF/CB]}} = \frac{\sigma_n^2}{P}\left(\sum_{i=1}^{K}\sqrt{\lambda_i}\right)^2$ one gets:

$$\begin{aligned} \mathrm{PSNR} &= 10\log_{10}\left(\frac{255^2 \cdot N}{\mathrm{D}_{\text{[LLSE/CB]}}}\right), \\ &= 20\log_{10}(255) \\ &- 10\log_{10}\left(\left(\frac{1}{\mathrm{CSNR}_{lin}+1}\right)\cdot\frac{1}{NK}\left(\sum_{i=1}^{K}\sqrt{\lambda_i}\right)^2\right) \\ &- 10\log_{10}\left(1+\frac{(\mathrm{CSNR}_{lin}+1)\cdot E_d}{{H_t}^2}\right), \end{aligned} \tag{A.3}$$

Therefore, the equation can be rewritten as:

$$\begin{aligned} \mathrm{PSNR} &= c + 10\log_{10}(\mathrm{CSNR}_{lin}+1) \\ &- 20\log_{10}(H_t) \\ &- 10\log_{10}\left(1+\frac{(\mathrm{CSNR}_{lin}+1)\cdot E_d}{{H_t}^2}\right), \\ &= c + \mathrm{CSNR} + 10\log_{10}\left(1+\frac{1}{\mathrm{CSNR}_{lin}}\right) \\ &- 20\log_{10}(H_t) \\ &- 10\log_{10}\left(1+\frac{(\mathrm{CSNR}_{lin}+1)\cdot E_d}{{H_t}^2}\right), \\ &= \mathrm{PSNR}_{\text{[LLSE/CB]*}}, \end{aligned} \tag{A.4}$$

where $10\log_{10}\left(1+\frac{1}{\mathrm{CSNR}_{lin}}\right)$ can be denoted by G_{LLSE} for ease of reading.

□

APPENDIX B
PROOF OF THE EQUATION OF THE SOFTCAST+ SCHEME

Proof. The proof of (57) is trivial when considering the fact that $\frac{\bar{P}+\sigma_n^2}{\sigma_n^2}$ can be seen as $\mathrm{CS\tilde{N}R}_{lin} = \mathrm{CSNR}_{lin} + 1$. Indeed, by inserting $\mathrm{CS\tilde{N}R}_{lin}$ in (24), replacing K by ℓ and using the property: $\log_{10}(a+b) = \log_{10}(a) + \log_{10}(1+\frac{b}{a})$, one easily gets:

$$\begin{aligned} \mathrm{PSNR} &= 10\log_{10}\left(\frac{255^2 \cdot N}{D_s + D_d}\right), \\ &= c - 10\log_{10}\left(1+\frac{D_d}{D_s}\right) + 10\log_{10}\left(\mathrm{CS\tilde{N}R}_{lin}\right) \\ &- 10\log_{10}\left(\frac{1}{N\ell}\left(\sum_{i=1}^{\ell}\sqrt{\lambda_i}\right)^2\right), \\ &= c + \mathrm{CSNR} + 10\log_{10}\left(1+\frac{1}{\mathrm{CSNR}_{lin}}\right) \\ &- 20\log_{10}(H_{t2}) - 10\log_{10}\left(1+\frac{\mathrm{CS\tilde{N}R}_{lin}\cdot E_{d2}}{{H_{t2}}^2}\right), \\ &= c + \mathrm{CSNR} + G_{\mathrm{LLSE}} - 20\log_{10}(H_{t2}) \\ &- 10\log_{10}\left(1+\frac{(\mathrm{CSNR}_{lin}+1)\cdot E_{d2}}{{H_{t2}}^2}\right), \\ &= \mathrm{PSNR}_{\text{[OPA-LLSE]}}. \end{aligned} \tag{B.1}$$

□

REFERENCES

[1] R. Xiong, F. Wu, J. Xu, X. Fan, C. Luo, and W. Gao, "Analysis of decorrelation transform gain for uncoded wireless image and video communication," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1820–1833, Apr. 2016.

[2] Cisco, "Cisco visual networking index: Global mobile data traffic forecast update, 2017-2022," Feb. 2019.

[3] I. E. G. Richardson, *The H.264 advanced video compression standard*, 2nd ed. Chichester: Wiley, 2010.

[4] G. J. Sullivan, J. R. Ohm, W. J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[5] S. Kokalj-Filipovi and E. Soljanin, "Suppressing the cliff effect in video reproduction quality," *Bell Labs Technical Journal*, vol. 16, no. 4, pp. 171–185, Mar. 2012.

[6] F. Liang, C. Luo, R. Xiong, W. Zeng, and F. Wu, "Superimposed Modulation for Soft Video Delivery with Hidden Resources," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 9, pp. 2345–2358, Sep. 2018.

[7] B. Tan, H. Cui, J. Wu, and C. W. Chen, "An Optimal Resource Allocation for Superposition Coding-Based Hybrid DigitalAnalog System," *IEEE Internet of Things Journal*, vol. 4, no. 4, pp. 945–956, Aug. 2017.

[8] Z. Zhang, D. Liu, and X. Wang, "Joint Carrier Matching and Power Allocation for Wireless Video with General Distortion Measure," *IEEE Transactions on Mobile Computing*, vol. 17, no. 3, pp. 577–589, Mar. 2018.

[9] C. Bachhuber, E. Steinbach, M. Freundl, and M. Reisslein, "On the Minimization of Glass-to-Glass and Glass-to-Algorithm Delay in Video Communication," *IEEE Transactions on Multimedia*, vol. 20, no. 1, pp. 238–252, Jan. 2018.

[10] M. A. Labiod, M. Gharbi, F.-X. Coudoux, P. Corlay, and N. Doghmane, "Cross-layer scheme for low latency multiple description video streaming over Vehicular Ad-hoc NETworks (VANETs)," *AEU - International Journal of Electronics and Communications*, vol. 104, pp. 23–34, May 2019.

[11] M. Wien, H. Schwarz, and T. Oelbaum, "Performance Analysis of SVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1194–1203, Sep. 2007.

[12] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian, "Overview of SHVC: Scalable Extensions of the High Efficiency Video Coding Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 20–34, Jan. 2016.

[13] T. Kratochvl, "Hierarchical Modulation in DVB-T/H Mobile TV Transmission," in *Multi-Carrier Systems & Solutions 2009*, ser. Lecture Notes in Electrical Engineering, S. Plass, A. Dammann, S. Kaiser, and K. Fazel, Eds. Dordrecht: Springer Netherlands, 2009, pp. 333–341.

[14] L. Yu, H. Li, and W. Li, "Wireless scalable video coding using a hybrid digital-analog scheme," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 2, pp. 331–345, 2014.

[15] S. Jakubczak and D. Katabi, "Softcast: one-size-fits-all wireless video," in *Proceedings of the ACM SIGCOMM 2010 conf.*, 2010, pp. 449–450.

[16] S. Zheng, M. Cagnazzo, and M. Kieffer, "Optimal and suboptimal channel precoding and decoding matrices for linear video coding," *Signal Processing: Image Communication*, vol. 78, pp. 135–151, Oct. 2019.

[17] T. Fujihashi, T. Koike-Akino, T. Watanabe, and P. V. Orlik, "High-Quality Soft Video Delivery With GMRF-Based Overhead Reduction," *IEEE Transactions on Multimedia*, vol. 20, no. 2, pp. 473–483, Feb. 2018.

[18] A. Trioux, G. Valenzise, M. Cagnazzo, M. Kieffer, F.-X. Coudoux, P. Corlay, and M. Gharbi, "Subjective and Objective Quality Assessment of the SoftCast Video Transmission Scheme," in *2020 IEEE International Conference on Visual Communications and Image Processing (VCIP)*, Dec. 2020, pp. 96–99.

[19] B. Tan, J. Wu, H. Cui, R. Wang, J. Wu, and D. Liu, "A Hybrid Digital Analog Scheme for MIMO Multimedia Broadcasting," *IEEE Wireless Communications Letters*, vol. 6, no. 3, pp. 322–325, Jun. 2017.

[20] J. Zhao, R. Xiong, C. Luo, F. Wu, and W. Gao, "Wireless image and video soft transmission via perception-inspired power distortion optimization," in *Proc. IEEE Visual Communications and Image Processing (VCIP)*, Dec. 2017, pp. 1–4.

[21] S. Jakubczak and D. Katabi, "SoftCast: Clean-slate scalable wireless video," *MIT Technical report*, Feb. 2011.

[22] Y. Gui, L. Hancheng, F. Wu, and C. W. Chen, "LensCast: Robust Wireless Video Transmission over mmWave MIMO with Lens Antenna Array," *IEEE Transactions on Multimedia*, pp. 1–1, 2020.

[23] Y. Gui, H. Lu, F. Wu, and C. W. Chen, "Robust Video Broadcast for Users with Heterogeneous Resolution in Mobile Networks," *IEEE Transactions on Mobile Computing*, pp. 1–1, 2020.

[24] X. Fan, R. Xiong, F. Wu, and D. Zhao, "Wavecast: Wavelet based wireless video broadcast using lossy transmission," in *Proc. IEEE Visual Communications and Image Processing (VCIP)*, Nov. 2012, pp. 1–6.

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

[26] Y. Wang, H. Lu, Z. Li, and J. Li, "Robust satellite image transmission over bandwidth-constrained wireless channels," in *Proc. IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–6.

[27] R. Xiong, J. Zhang, F. Wu, J. Xu, and W. Gao, "Power Distortion Optimization for Uncoded Linear Transformed Transmission of Images and Videos," *IEEE Transactions on Image Processing*, vol. 26, no. 1, pp. 222–236, Jan. 2017.

[28] S. Zheng, M. Antonini, M. Cagnazzo, L. Guerrieri, M. Kieffer, I. Nemoianu, R. Samy, and B. Zhang, "Softcast with per-carrier power-constrained channels," in *Proc. IEEE International Conference on Image Processing (ICIP)*, Aug. 2016, pp. 2122–2126.

[29] G. Baruffa and F. Frescura, "Performance of SoftCast and H.265 in software radio video multicasting systems," in *2017 International Symposium on Wireless Communication Systems (ISWCS)*, Aug. 2017, pp. 25–30.

[30] X.-W. Tang and X.-L. Huang, "A Design of SDR-Based Pseudo-analog Wireless Video Transmission System," *Mobile Networks and Applications*, vol. 25, pp. 2495–2505, 2020.

[31] Kyong-Hwa Lee and D. Petersen, "Optimal Linear Coding for Vector Channels," *IEEE Transactions on Communications*, vol. 24, no. 12, pp. 1283–1290, Dec. 1976.

[32] J. Wu, J. Wu, H. Cui, C. Luo, X. Sun, and F. Wu, "DAC-Mobi: Data-Assisted Communications of Mobile Images with Cloud Computing Support," *IEEE Transactions on Multimedia*, vol. 18, pp. 893–904, May 2016.

[33] A. Trioux, F.-X. Coudoux, P. Corlay, and M. Gharbi, "Temporal information based gop adaptation for linear video delivery schemes," *Signal Processing: Image Communication*, vol. 82, p. 115734, 2020.

[34] H. Cui, C. Luo, C. W. Chen, and F. Wu, "Robust uncoded video transmission over wireless fast fading channel," in *INFOCOM, 2014 Proceedings IEEE*. IEEE, 2014, pp. 73–81.

[35] D. He, C. Lan, C. Luo, E. Chen, F. Wu, and W. Zeng, "Progressive Pseudo-analog Transmission for Mobile Video Streaming," *IEEE Transactions on Multimedia*, vol. 19, no. 8, pp. 1894–1907, Aug. 2017.

[36] Y. Li, Z. Li, Y. Liu, and Y. Wang, "SCAST: Wireless Video Multicast Scheme Based on Segmentation and Softcast," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, 2017, pp. 1–6.

[37] W. Yin, X. Fan, and Y. Shi, "Convolutional Neural Networks Based Soft Video Broadcast," in *Advances in Multimedia Information Processing PCM 2018*, ser. Lecture Notes in Computer Science, R. Hong, W.-H. Cheng, T. Yamasaki, M. Wang, and C.-W. Ngo, Eds. Springer International Publishing, 2018, pp. 641–650.

[38] F. Liang, C. Luo, R. Xiong, W. Zeng, and F. Wu, "Hybrid DigitalAnalog Video Delivery With ShannonKotelnikov Mapping," *IEEE Transactions on Multimedia*, vol. 20, no. 8, pp. 2138–2152, Aug. 2018.

[39] T. Zhang and S. Mao, "Metadata Reduction for Soft Video Delivery," *IEEE Networking Letters*, pp. 84–88, Apr. 2019.

[40] S. Zong, S. Gao, G. Tu, C. Zhang, and D. Chen, "A metadata-free pure soft broadcast scheme for image and video transmission," *Signal Processing: Image Communication*, Jul. 2019.

[41] D. Salomon and G. Motta, *Handbook of Data Compression*, 5th ed. London: Springer-Verlag, 2010.

[42] J. G. Yim, Y. Wang, N. Birkbeck, and B. Adsumilli, "Subjective Quality Assessment For Youtube Ugc Dataset," in *2020 IEEE International Conference on Image Processing (ICIP)*, Oct. 2020, pp. 131–135, iSSN: 2381-8549.

[43] "Xiph.org media." [Online]. Available: https://media.xiph.org/video/derf/

[44] M. Wien, *High Efficiency Video Coding: Coding Tools and Specification*, ser. Signals and Communication Technology. Berlin Heidelberg: Springer-Verlag, 2015.

[45] J. Shen, L. Yu, L. Li, and H. Li, "Foveation-Based Wireless Soft Image Delivery," *IEEE Transactions on Multimedia*, vol. 20, no. 10, pp. 2788–2800, Oct. 2018.

[46] P. ITU-T RECOMMENDATION, "Subjective video quality assessment methods for multimedia applications," Sep. 1999.

[47] T. Brandao, L. Roque, and M. P. Queluz, "Quality assessment of H.264/AVC encoded video," in *Proc. of Conference on Telecommunications - ConfTele*, Sta. Maria da Feira, Portugal, 2009, p. 5.

[48] M. Frigo and S. Johnson, "The Design and Implementation of FFTW3," *Proceedings of the IEEE*, vol. 93, no. 2, pp. 216–231, Feb. 2005.