On the Impact of Lossy Channels in Wireless Edge Caching

Angela Sara Cacciapuoti*, Marcello Caleffi*, Mingyue Ji†, Jaime Llorca‡, Antonia Maria Tulino*‡
*DIETI Department, University of Naples Federico II, Italy. Email: {angelasara.cacciapuoti, marcello.caleffi}@unina.it
† Broadcom Limited, CA. Email: {mingyue.ji@broadcom.com}
‡ Bell Labs, Nokia, NJ. Email: {jaime.llorca, a.tulino}@nokia.com

***Abstract*—One of the main challenges for continued wireless capacity growth is the difficulty in exploiting the multicast nature of the wireless medium: wireless end points rarely experience the same channel conditions or access the same content at the same time. In this paper, we present and analyze a novel wireless video delivery paradigm based on the combined use of channel-aware caching and coded multicasting that allows simultaneously serving multiple cache-enabled access points that may be requesting different content and experiencing different channel conditions. To this end, we reformulate the caching-aided coded multicast problem as a joint source-channel coding problem and design an achievable scheme that preserves the cache-enabled multiplicative throughput gains of the error-free scenario, by guaranteeing per-receiver (access point) rates unaffected by the presence of receivers with worse channel conditions.**

***Index Terms*—Video delivery, coded multicast, caching, wireless channel, joint source-channel coding**

I. Introduction

The latest projections [2], [3] suggest that, by 2019, mobile data traffic will increase nearly tenfold with respect to 2014, with mobile traffic accounting for nearly two-thirds of the total data traffic by 2018. Furthermore, it is suggested that nearly three-fourths of the mobile data traffic will be video by 2019. In line with these trends, this paper considers the design and analysis of a novel wireless video delivery paradigm that specifically addresses two of the major predicted shifts of the Future Internet, i.e., *from fixed to mobile* and *from cable video consumption to IP video consumption*, by pushing caching to the wireless edge and exploiting the multicast nature of the wireless medium via channel-aware coded multicasting.

In this setting, recent information theoretic studies have shown that the use of wireless edge caching and coded multicasting over the wireless backhaul can significantly improve the performance of wireless caching networks by creating coded multicast transmissions useful for multiple users (access points) even if requesting different content [4]–[11]. However, the underlying assumption in existing information theoretic literature on caching networks is that the channels between the content source and the users exhibit the same qualities and follow a shared error-free deterministic model. In practice, wireless channels are affected by impairments, such as multipath and shadow fading, and they must be modeled as non-deterministic noisy channels. Furthermore, different wireless channels can exhibit significant differences in their qualities, i.e., in their admissible throughputs, even when the receivers are closely located. This channel condition heterogeneity can significantly degrade the performance of any multicast-based approach in which the worst-channel user dictates the overall performance [12], [13].

In the following, we address these open problems by considering a heterogeneous wireless edge architecture composed of a macro-cell Base Station (BS) that distributes video content to a number of mobile devices with the help of dedicated cache-enabled femto/pico cell base stations referred to as *helpers*. Specifically, by exploiting the helper caching capabilities, the BS distributes the video content through coded multicast transmissions, reducing both content distribution latencies as well as overall network load [8], [14].

We formulate the channel-aware caching-aided coded multicast problem as a *joint source-channel coding problem* providing the following key contributions:

i) An information theoretic framework for wireless video distribution that takes into account the specific characteristics of the wireless propagation channel in the presence of any combination of unicast/multicast transmission and wireless edge caching;
ii) A channel-aware caching-aided coded multicast video delivery scheme, referred to as Random Popularity based caching and Channel-Aware Chromatic-number Index Coding (RAP-CA-CIC), that guarantees the highest admissible video throughput to each helper for the given propagation conditions, i.e., completely avoiding throughput penalizations from the presence of helpers experiencing worse propagation conditions;
iii) A novel polynomial-time approximation of RAP-CA-CIC, referred to as Random Popularity based caching and Channel-Aware Hierarchical greedy Coloring (RAP-CA-HgC) with running time quadratic in both the number of receivers (helpers) and the number of (per-receiver) requested video descriptions.

This work was supported in part by the PON projects "FERSAT: studio di un sistema di segnalamento FERroviario basato sull'innovativo utilizzo delle tecnologie SATellitari e della loro integrazione con le tecnologie terrestri" and DATABANC "CHIS: Cultural Heritage Information System", and in part by the Campania POR project "myOpenGov".

A longer version of this paper, including all the proofs, has been published in the IEEE Journal of Selected Areas in Communications [1].

978-1-4799-6664-6/16/$31.00 ©2016 IEEE

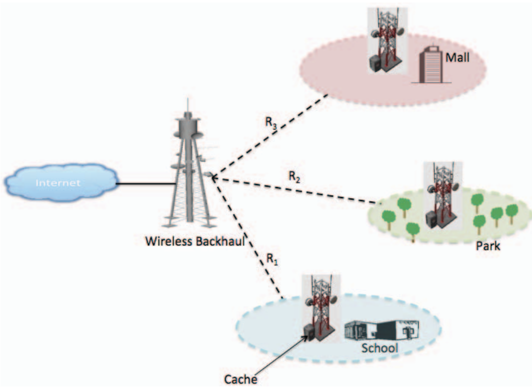

Fig. 1. Network Model

The rest of the paper is organized as follows. Section II presents the network model. Section III formalizes the video delivery problem via a general information-theoretic formulation. In Section IV, we first specialize the aforementioned information-theoretic formulation for a specific choice of the cache encoder and of the joint source-channel multicast encoder, leading to the design of the RAP-CA-CIC scheme. We then further provide the polynomial-time algorithm RAP-CA-HgC. Section V presents simulation results of the proposed schemes and related discussions. We conclude the paper in Section VI.

II. Network Model

We consider a wireless single-hop broadcast caching network consisting of a sender node (base station) and U receiver nodes (helpers) $\mathcal{U} = \{1, \ldots, U\}$. The sender has access to a content library $\mathcal{F} = \{1, \ldots, m\}$ containing m files, where each file has entropy equal to F bits. Each receiver $u \in \mathcal{U}$ has a cache with storage capacity $M_u F$ bits (i.e., M_u files). Without loss of generality, the files are represented by binary vectors $W_f \in \mathbb{F}_2^F$.

Differently from existing works for this network [4]–[10], here the sender is connected to the receivers via lossy links. Specifically, we model the links between the sender and the receivers as stochastically degraded binary broadcast channels (BC). Without loss of generality, we denote by $\epsilon_u, u \in \mathcal{U}$ the channel degradation of the u-th link.

We consider a video streaming application, in which each file $f \in \mathcal{F}$ represents a video segment, which is multiple description coded into D descriptions, using, for example, one of the coding schemes described in [12], [15]. Each description is packaged into one information unit or packet for transmission. Each packet is represented as a binary vector of length (entropy) $B = F/D$ bits. A low-quality version of the content can be reconstructed once a receiver is able to recover any description. The reconstruction quality improves by recovering additional descriptions and depends solely on the number of descriptions recovered, regardless of the particular recovered collection. Hence, there are D video qualities per segment, where the entropy in bits of quality $d \in \{1, \ldots, D\}$, containing d descriptions, is given by $F_d = B\,d$. Note that in video streaming applications, each video segment has the same (playback) duration, which we denote by Δ in time units (or channel uses)[1]. Hence, the difference in quality levels solely depend on the number of bits F_d.

Video segments are characterized by a *demand distribution* $\mathbf{Q} = [q_{f,u}], u = 1, \cdots, U, f = 1, \cdots, m$, where $q_{f,u} \in [0, 1]$ and $\sum_{f=1}^{m} q_{f,u} = 1$ (e.g., receiver u requests file f with probability $q_{f,u}$). Without loss of generality up to index reordering, we assume $q_{f,u}$ has non-increasing components $q_{1,u} \geq \cdots \geq q_{m,u}$. Let f_u denote the random request at receiver u. The realization of f_u is denoted by f_u.

We consider a video delivery system that operates in two phases: a caching (or placement) phase followed by a transmission phase.

- **Caching Phase:** The caching phase occurs during a period of low network traffic. In this phase, using the knowledge of the demand distribution and the cache capacity constraints, the sender decides what content to store at each receiver. We denote by $\mu_{u,f}$ the number of descriptions of file f cached at receiver u.
- **Transmission Phase:** After the caching phase, the network is used in a time slotted fashion with time-slot duration Δ time units, given by the video streaming application. We assume each receiver requests one video segment per time-slot. Based on the receiver requests, the stored contents at the receiver caches, and the channel conditions, the sender decides at what quality level to send the requested files, encodes the chosen video segments into a codeword, and sends it over the shared link to the receivers, such that every receiver decodes the requested segment (at the corresponding quality level) within Δ time units. We denote by d_u the scheduled quality level for receiver u, i.e., the number of descriptions scheduled for receiver u.

III. Problem Formulation

In the following, we introduce a general information-theoretic formulation of the wireless video delivery problem described above. As already mentioned, we model the files by binary vectors $W_f \in \mathbb{F}_2^F$ of entropy F. At the beginning of time, a realization of the library $\{W_f\}$[2] is revealed to the sender. A $(\Delta, \{\epsilon_u\})$-delivery scheme consists of:

Cache Encoder:
Given the knowledge of the demand distribution $\mathbf{Q}$ and the cache sizes $\{M_u\}$, the sender fills the caches of the U receivers through a set of U encoding functions $\{Z_u : \mathbb{F}_2^{mF} \rightarrow \mathbb{F}_2^{M_u F} : u \in \mathcal{U}\}$, such that $Z_u(\{W_f\})$denotes the codeword stored in the cache of receiver u.

Multicast Encoder:
Given the knowledge of network time-slot duration Δ, the cache contents $\{Z_u\}$, the channel conditions $\{\epsilon_u\}$, and the

[1]In the rest of the paper we use "time units" and "channel uses" interchangeably.

[2]For ease of exposition, in the following, unless specified, we denote by $\{A_i\}$ the full set of elements $\{A_i : i \in \mathcal{I}\}$.

receiver requests, the sender schedules the appropriate quality level for each video segment request through a multicast encoder which is defined by a variable-to-fixed encoding function $\mathbf{X} : \mathbb{F}_2^{mF} \times \mathcal{F}^U \to \mathbb{F}_2^{\Delta}$ (where $\mathbb{F}_2^{\Delta}$ denotes the set of finite length binary sequences)[3], such that the transmitted codeword is given by

$$\mathbf{X} \triangleq \mathbf{X}(\{W_f\}, \{\epsilon_u\}, \mathbf{f}), \tag{1}$$

where $\mathbf{f} \triangleq [f_1, f_2, \ldots, f_U]$, with $f_j \in \mathcal{F}$, denotes the realization of the receiver random request vector, $\mathsf{f} = [\mathsf{f}_1, \mathsf{f}_2, \ldots, \mathsf{f}_U]$.

Multicast Decoders:
Having observed its channel output $\mathbf{Y}_u$, each receiver decodes the requested file $W_u \in \mathcal{F}$ as $\widehat{W}_u = \lambda_u(\mathbf{Y}_u, Z_u, \mathbf{f})$, where $\lambda_u : \mathcal{Y}^{\Delta} \times \mathbb{F}_2^{M_u F} \times \mathcal{F}^U \to \mathbb{F}_2^F$ denotes the decoding function of receiver u (where $\mathcal{Y}^{\Delta}$ denotes the set of the received sequences). The worst-case (over the file library)[4] probability of error of the $(\Delta, \{\epsilon_u\})$-delivery scheme over a stochastically degraded broadcast channel is defined as

$$P_e^{(F)}(\mathbf{X}, \{Z_u\}, \{\lambda_u\}, \{\epsilon_u\}) = \sup_{\{W_f\}} \mathbb{P}\left(\bigcup_{u \in \mathcal{U}} \left\{ \widehat{W}_u \neq W_u \right\} \right). \tag{2}$$

Definition 1. A sequence of $(\Delta, \{\epsilon_u\})$-delivery schemes is called *admissible* if

$$\lim_{F \to \infty} P_e^{(F)}(\mathbf{X}, \{Z_u\}, \{\lambda_u\}, \{\epsilon_u\}) = 0.$$

Remark 1. Note that the variable-to-fixed nature of the encoder is due to that fact that while the length of the transmitted codeword $\mathbf{X}$ is fixed and given by Δ, the amount of information bits transmitted by the encoder is variable, given by $\sum_u d_u B$.

Remark 2. Note that the total video quality delivered to receiver u is given by $d_u + \mu_{u,f}$, i.e., the number of descriptions delivered via the transmitted codeword plus the number of descriptions delivered from the local cache.

IV. Random Fractional Caching and Channel-Aware Chromatic Index Coding

Here, we focus on a particular class of admissible $(\Delta, \{\epsilon_u\})$-delivery schemes based on a specific choices of **Cache Encoder** and joint source-channel **Multicast Encoder**. Specifically, we consider a cache encoder based on RAndom fractional Popularity-based (RAP) caching [7], while for the joint source-channel multicast encoder, we design a new transmission scheme based on channel-aware graph coloring. In the following, for ease of notation, we assume that $M_u = M$ and $q_{f,u} = q_f$ for all $u \in \mathcal{U}$.[5]

[3] The symbol $\mathbb{F}_2^{\Delta}$ is used to indicate that the codeword length is fixed and dictated by the duration time-slot duration Δ.

[4] $\sup_{\{W_f\}}(\cdot)$ in (2) considers the worst-case (over the file library).

[5] Results for the heterogenous scenario are included in [1].

A. Cache Encoder

The caching phase explores the fact that video segments are multiple description coded into D descriptions, each of length $B = F/D$ bits. In the following, such descriptions are referred to as packets and denoted by $\{W_{f,\ell} : \ell = 1, \ldots, D\}$. Specifically, the caching phase works as follows:

- Each receiver, instructed by the sender, randomly selects and stores in its cache a collection of $p_f MD = M_f D$ distinct packets (descriptions) from each file $f \in \mathcal{F}$, where $\mathbf{p} = (p_1, \ldots, p_m)$ is a vector with components $0 \leq p_f \leq 1/M, \forall f$, such that $\sum_{f=1}^m p_f = 1$, referred to as the caching distribution of receiver u. Hence, the arbitrary element p_f of $\mathbf{p}$ represents the fraction of the memory M allocated to file f.

The collection of cached packet indices over all receivers and all files is a random vector denoted in the following by M, while a given cache configuration, i.e., a realization of M is denoted by $\mathbf{M}$. Moreover, we denote by $\mathbf{M}_{u,f}$ the vector of indices of the packets of file f cached by receiver u whose cardinality $|\mathbf{M}_{u,f}|$ is given, based on the above assumption, by $p_f MD$. The aforementioned caching policy, referred to as *RAP caching*, is completely characterized by the caching distribution $\mathbf{p}$ and it is synthesized in Algorithm 1.

Algorithm 1 RAndom fractional Popularity-based caching

```
1: for all f ∈ F do
2:     Each receiver u caches a subset (M_{u,f}) of p_f MD distinct
       packets of file f uniformly at random.
3: end for
4: return M = [M_{u,f}], u ∈ U, f ∈ F.
```

B. Multicast Encoder

The proposed joint source-channel multicast encoder is composed of:

- A *Channel-aware Source Compressor*;
- An *Inner-Outer Channel-aware Source Encoder*.

Recall that the joint source-channel multicast encoder is a variable-to-fixed encoder where, at each realization of the request vector $\mathbf{f}$, the number of descriptions d_u scheduled by the sender for each receiver u is set to

$$d_u \frac{F}{D} = \Delta \frac{\eta_u}{\bar{\chi}}, \tag{3}$$

where

$$\bar{\chi} = \min\{\psi(\mathbf{p}, \mathbf{Q}), \bar{m}\},$$

$$\psi(\mathbf{p}, \mathbf{Q}) = \sum_{f=1}^{m} \sum_{\ell=1}^{U} \binom{U}{\ell} \rho_{f,\ell} (1 - p_f M)^{U-\ell+1} (p_f M)^{\ell-1},$$

$$\rho_{f,\ell} \triangleq \mathbb{P}(f = \operatorname*{argmax}_{j \in \Upsilon} \ (p_j M)^{\ell-1} (1 - p_j M)^{U-\ell+1}),$$

with Υ a random set of ℓ elements selected in an i.i.d. manner from $\mathcal{F}$ (with replacement) and $\bar{m} = \sum_{f=1}^m \left(1 - (1 - q_f)^U\right)$.

In [1], we show that $\bar{\chi}$ represents the average, over the demand vector f and caching realization M, of the *Channel-Aware Chromatic Number* (see Definition 4 in Section IV-D). Note that $\bar{\chi}$ depends only on the system parameters (i.e., caching distributions, cache sizes, demand distributions, number of receivers, and library size).

1) Channel-aware Source Compressor: The goal of the channel-aware source compressor is to cluster the set of packets (descriptions) scheduled for each receiver $\{d_u\}$ into a smaller set of equivalent classes. A key concept driving this process is what we refer to as *generalized independent set* (GIS). As shall be clear from its formal definition in Section IV-D, the concept of GIS generalizes the classical notion of independent set in the graph coloring literature. The proposed compressor clusters the entire set of scheduled packets (descriptions) into the minimum number of GISs satisfying the following conditions:

- Any two packets in a GIS scheduled for different receivers can be transmitted in the same time-frequency slot without affecting decodability.
- The set of packets in a GIS scheduled for receiver u, denoted by $\mathcal{P}_u$, satisfies
$$\frac{|\mathcal{P}_u|}{\eta_u} = \frac{|\mathcal{P}_i|}{\eta_i}, \quad \forall u, i \in \{1, \ldots, U\}. \tag{4}$$
Note that (4) makes sure that each receiver is scheduled a number of packets (descriptions) proportional to its channel rate.

As described in Section IV-D, this minimization corresponds to a NP-hard optimization problem related to finding the minimum number of GISs that cover a properly constructed *conflict graph*. We refer to the minimum number of GISs needed to cover the conflict graph as *channel-aware chromatic-number*, χ_{CA}, and to the associated transmission scheme as *Channel-Aware Chromatic-number Index Coding (CA-CIC)* (see Section IV-D). Given the exponential complexity of CA-CIC, we then provide in Section IV-E a practical polynomial-time approximation of CA-CIC, referred to as *Channel-Aware Hierarchical greedy Coloring (CA-HgC).*

2) Concatenated Channel-Source Encoder: The goal of the concatenated source-channel encoder is to generate the coded multicast codeword $\mathbf{X}$. To this end, the encoder, first, takes as input the GISs generated by the channel-aware source compressor, and encodes the descriptions associated to each set $\mathcal{P}_u$ belonging to a given GIS using the codebook of receiver u, $\mathcal{C}_u = \{\mathcal{C}_u(j) : j \in \{1, \ldots, 2^{n\eta_u}\}\}$, generated according to its channel rate η_u. The codebooks of all the users are characterized by a common codeword length $n = \Delta/\bar{\chi}$.[6] After the generation of the per-receiver channel codewords, the final coded multicast codeword is generated by XORing the channel codewords belonging to the same GIS and concatenating the resulting XORed codewords. As shown in [1], the average length (over the demand vector f) of the resulting multicast codeword is equal to Δ.

[6]We remark that the sender notifies the computed codebooks to each receiver at network setup or anytime channel conditions change. Hence, each receiver is aware, not only of its own codebook, but also of the codebooks of the other receivers.

The decodability of the constructed multicast codeword is described in the following subsection.

C. Multicast Decoder

From the observation of its channel output $\mathbf{Y}_u$, representing its noisy version of the transmitted codeword $\mathbf{X}$, each receiver is able to decode the d_u descriptions of its requested file, $W_u \in \mathcal{F}$, scheduled and transmitted by the sender, as $\widehat{W}_u = \lambda_u(\mathbf{Y}_u, Z_u, \mathbf{f})$, via its own decoding function, λ_u, which consists of two stages:

- First, receiver u is informed (e.g., via packet header information) of the sub-codewords in the concatenated multicast codeword $\mathbf{X}$ that contain any of its scheduled descriptions. For each of the identified sub-codewords, receiver u obtains the noisy version of its channel codeword $\mathfrak{C}_u(j)$, $j \in \{1, \ldots, 2^{n\eta_u}\}$, by performing the inverse XOR function. To this end, receiver u is informed of the packets and their intended receivers that are present in the XORed sub-codeword. Receiver u can then construct the channel codewords associated to the other receivers from its cached information and the corresponding codebooks (recall that every receiver is informed of all the channel codebooks) and recover its own codeword $C_u(j)$ via inverse XORing.
- Then, the recovered noisy codeword $C_u(j)$ is sent to the channel decoder of receiver u, which reconstructs the bits associated to the subset of scheduled descriptions $\mathcal{P}_u$.

Hence, according to Def. 1, in the limit $F \to +\infty$, the proposed sequence of $(\Delta, \{\epsilon_u\})$-delivery schemes is *admissible.*

D. Channel-Aware CIC

The request of a file f_u by receiver u translates into the request of $D - |\mathbf{M}_{u,f}|$ packets. The sender schedules the quality level for receiver u by choosing a number of descriptions $d_u \in \{1, \ldots, D(1 - M_{u,f})\}$ packets. In line with the notation introduced for the cache configuration, we denote by $\mathbf{W}$ the scheduled packet-level configuration and by $\mathbf{W}_{u,f}$ the set of packets of file f scheduled for receiver u.

Following the index coding literature [7], [16], we define the conflict graph $\mathcal{H}_{\mathbf{M},\mathbf{W}} = (\mathcal{V}, \mathcal{E})$ corresponding to the index coding problem defined by $(\mathbf{M}, \mathbf{W})$ as follows:

- Consider each packet of size B requested by one receiver as a single vertex in $\mathcal{H}_{\mathbf{M},\mathbf{W}}$. Hence, each vertex $v \in \mathcal{V}$ is uniquely identified by the pair $\{\rho(v), \mu(v)\}$ where $\rho(v)$ indicates the packet identity associated to the vertex v and $\mu(v)$ represents the receiver requesting it. In total, we have $|\mathcal{V}| = \sum_{u \in \mathcal{U}} d_u$ vertices.
- For any pair of vertices v_1, v_2, we say that vertex (packet) v_1 interferes with vertex v_2 if the packet associated to vertex v_1, $\rho(v_1)$, is not in the cache of the receiver associated to vertex v_2, $\mu(v_2)$, and $\rho(v_1)$ and $\rho(v_2)$ do not represent the same packet. Then, draw an edge between vertex v_2 and vertex v_1, if v_1 interferes with v_2 or v_2 interferes with v_1.

Within such conflict graph $\mathcal{H}_{\mathbf{M},\mathbf{W}}$, we define $\mathcal{V}_u \in \mathcal{V}$ as the set of vertices in $\mathcal{V}$ such that $\mu(v) = u$. Such $\{\mathcal{V}_u\}$ are fully-connected subgraphs of $\mathcal{V}$. Based on this consideration, we refer to $\mathcal{H}_{\mathbf{M},\mathbf{W}}$, as a U-clustered graph.

Starting from the constructed conflict graph $\mathcal{H}_{\mathbf{M},\mathbf{W}}$, a new concept of independent set has to be defined, which we refer to as Generalized Independent Set (GIS) of a U-clustered graph.

Definition 2. We define a $(s_1, \ldots, s_U)$-GIS of a U-clustered graph $\mathcal{H}_{\mathbf{M},\mathbf{W}}$ as a set of U fully connected sub-graphs $\{\mathcal{P}_1, \ldots \mathcal{P}_U\}$ such that for all $u = 1, \ldots, U$:

- For all $v \in \mathcal{P}_u$, $\mu(v) = u$ (i.e., all the packets in $\mathcal{P}_u$ are scheduled for receiver u)
- $|\mathcal{P}_u| = s_u \geq 0$ (i.e., the number of packets in $\mathcal{P}_u$ is equal to s_u)
- For all $i \neq u$, $\mathcal{P}_u$ and $\mathcal{P}_i$ are mutually disconnected (i.e., no edges exist between any two subgraphs)

Note that when $s_u \leq 1, \forall u \in \mathcal{U}$, Def. 2 becomes the classical definition of independent set. Based on the notion of GIS, we introduce the definition of channel-aware valid coloring and channel-aware chromatic number:

Definition 3. (**Channel-Aware Valid Vertex-Coloring**) A $(\eta_1, \ldots, \eta_U)$ channel-aware valid vertex-coloring is obtained by covering the conflict graph $\mathcal{H}_{\mathbf{M},\mathbf{W}}$ with $(s_1, \ldots, s_U)$-GISs under the constraints (4), and assigning the same color to all the vertices in the same GIS.

Definition 4. (**Channel-Aware Chromatic Number**) The $(\eta_1, \ldots, \eta_U)$ channel-aware chromatic number of a graph $\mathcal{H}$ is defined as

$$\chi_{\text{CA}}(\mathcal{H}) = \min_{\{\mathcal{C}\}} |\mathcal{C}|, \tag{5}$$

where $\{\mathcal{C}\}$ denotes the set of all $(\eta_1, \ldots, \eta_U)$ channel-aware valid vertex-colorings of $\mathcal{H}$, and $|\mathcal{C}|$ is the total number of colors in $\mathcal{H}$ for the given $(\eta_1, \ldots, \eta_U)$ channel-aware valid vertex-coloring $\mathcal{C}$.

Theorem 1. *Given a conflict graph $\mathcal{H}_{\mathbf{M},\mathbf{W}}$ constructed according to packet-level cache realization $\mathbf{M}$, demand realization $\mathbf{f}$, and scheduled random packet-level configuration $\mathbf{W}$, a tight upper-bound for the channel-aware chromatic number $\chi_{CA}(\mathcal{H}_{\mathbf{M},\mathbf{W}})$, when $D, \Delta \to \infty$, is given by $\psi(\mathbf{f}, \mathbf{M})$, i.e.,*

$$\chi_{CA}(\mathcal{H}_{\mathbf{M},\mathbf{W}}) = \psi(\mathbf{f}, \mathbf{M}) + o(1/D). \tag{6}$$

Proof: See [1]. ■

Example 1. *Consider a network with $U = 3$ receivers, denoted by $\mathcal{U} = \{1, 2, 3\}$, and $m = 3$ files, denoted by $\mathcal{F} = \{W_a, W_b, W_c\}$. The channel rates are $\eta_1 = \frac{1}{2}$, $\eta_2 = \frac{1}{4}$, $\eta_3 = \frac{1}{4}$, and the time-slot duration is $\Delta = 8$ channel uses. We assume that the demand and caching distributions are such that $\bar{\chi} = 2$. Each file is multiple description coded into D descriptions, e.g., $W_a = \{W_{a,1}, W_{a,2}, \ldots, W_{a,D}\}$, each of length $B = F/D = 1$ bits. According to Eq.* (3)*, the sender schedules 2 descriptions for receiver 1, i.e., $d_1 = 2$, and one description for both receivers 2 and 3, i.e., $d_2 = d_3 = 1$. Assume that the caching realization given by: $\mathbf{M}_{1,W_a} = \{W_{a,1}, W_{a,2}\}$, $\mathbf{M}_{1,W_c} = \{W_{c,1}\}$; $\mathbf{M}_{2,W_b} = \{W_{b,1}, W_{b,2}\}$, $\mathbf{M}_{2,W_c} = \{W_{c,1}\}$; $\mathbf{M}_{3,W_a} = \{W_{a,1}\}$, $\mathbf{M}_{3,W_b} = \{W_{b,1}, W_{b,3}\}$. Suppose receiver 1 requests W_b, receiver 2 requests W_a, and receiver 3 requests W_c such that $\mathbf{W}_{1,W_b} = \{W_{b,1}, W_{b,2}\}$, $\mathbf{W}_{2,W_a} = \{W_{a,1}\}$ and $\mathbf{W}_{3,W_c} = \{W_{c,1}\}$. In Figure 2, we show two (s_1, s_2, s_3)-GISs covering the* 3*-clustered graph $\mathcal{H}_{\mathbf{M},\mathbf{W}}$. Recall that each vertex v in $\mathcal{H}_{\mathbf{M},\mathbf{W}}$ is identified by the pair $\{\rho(v), \mu(v)\}$ with $\rho(v)$ indicating the packet identity and $\mu(v)$ the requesting receiver. The first GIS is composed of two nonempty fully connected sub-graphs: $\mathcal{P}_1 = \{v_1, v_2\}$ with $s_1 = |\mathcal{P}_1| = 2$, and $\mathcal{P}_2 = \{v_3\}$ with $s_2 = |\mathcal{P}_2| = 1$. The second GIS is composed of one non-empty fully connected subgraph: $\mathcal{P}_3 = \{v_4\}$, with $s_3 = |\mathcal{P}_3| = 1$. The channel-aware vertex coloring is depicted in Fig. 2, with channel-aware chromatic number equal to two colors, one for each GIS (red dotted circle and black dotted circle). The multicast codeword $\mathbf{X}$ of length $\Delta = 8$ is obtained concatenating the codewords $\mathcal{C}_1(\mathcal{P}_1) \oplus \mathcal{C}_2(\mathcal{P}_2)$ and $\mathcal{C}_3(\mathcal{P}_3)$, each of length $n = 4$, where $\mathcal{C}_u(\mathcal{P}_u)$ is the channel codeword associated to the scheduled packets in $\mathcal{P}_u$.*

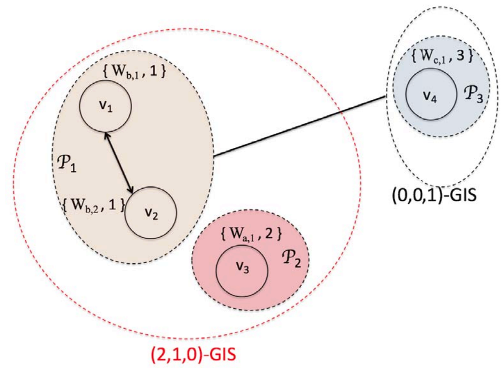

Fig. 2. An example of (s_1, s_2, s_3)-GISs of the 3-clustered graph $\mathcal{H}_{\mathbf{M},\mathbf{W}}$.

It is important to note that computing the chromatic number of a graph is a well known NP-Hard problem. In the next subsection, we introduce a polynomial-time approximation of CA-CIC, referred to as *Channel-Aware Hierarchical greedy Coloring (CA-HgC)*.

E. Channel-Aware Hierarchical greedy Coloring (CA-HgC)

We now present CA-HgC, a novel coded multicast algorithm that fully accounts for the broadcast channel impairments, while exhibiting polynomial-time complexity. CA-HgC generalizes the hierarchical greedy coloring (HgC) algorithm proposed in [9] for the error-free shared link scenario.

The CA-HgC algorithm works by computing two valid colorings of the conflict graph $\mathcal{H}_{\mathbf{M},\mathbf{W}}$, referred to as CA-HgC$_1$ and CA-HgC$_2$. CA-HgC then compares the corresponding number of colors achieved by the two solutions and selects the coloring with minimum number of colors. Specifically, CA-HgC$_1$ coincides with the naive (uncoded) multicasting. In fact, CA-HgC$_1$ computes the minimum coloring of $\mathcal{H}_{\mathbf{M},\mathbf{W}}$

subject to the constraint that only the vertices representing the same packet are allowed to have the same color. In this case, the total number of colors is equal to the number of distinct requested packets, and the coloring can be found in $O(|\mathcal{V}|^2)$. On the other hand, CA-HgC$_2$ is described by Algorithm 2, in which the subroutine `GISfunction`$(\cdot,\cdot,\cdot)$ is defined by Algorithm 3. It can be shown that the complexity of CA-HgC$_2$ is given by $O(U|\mathcal{V}|^2)$, i.e., it is polynomial in $|\mathcal{V}|$. In the following, we first guide the reader through Algorithm 2[7], then we provide an example to clarify the procedure.

Let $\mathcal{K}_v \triangleq \{u \in \mathcal{U} : \rho(v) \in \mathbf{W}_u \cup \mathbf{M}_u\}$ and $\mathcal{H}_i \triangleq \{v : |\mathcal{K}_v| = i\}$. We consider $\mathcal{H}_i$ to represent the i-th hierarchy. For CA-HgC$_2$, we start from the U-th hierarchy. For each vertex v in the i-th hierarchy ordered according to the vertex cardinality $\mathcal{K}_v$, we call the subroutine *GISfunction*(H_i, v, i). If the subroutine returns a non-empty set $\mathcal{G}$, then we color the vertices in $\mathcal{G}$ (with $v \in \mathcal{G}$) with the same color (lines 6-10). Otherwise, we move the uncolored vertex v to the next hierarchy, i.e., $\mathcal{H}_{i-1}$ (lines 11-12). This procedure is iteratively applied for any hierarchy, until all the vertices in the conflict graph $\mathcal{H}_{\mathbf{M},\mathbf{W}}$ are colored, and the number of color $|\mathcal{A}|$ as well as the set of codewords $\mathbf{X}$ are returned. As regards to subroutine *GISfunction*(H_i, v, i) shown in Algorithm 3, we set initially $\mathcal{G} = \{v\}$ and we start collecting in $\mathcal{G}$ all the vertices in $\mathcal{H}_i$ with lowest-cardinality having no links between each others within the conflict graph (lines 2-8). Note that at the first iteration the lowest cardinality in $\mathcal{H}_i$ is exactly equal to i. By construction, each vertex in the conflict graph has an associated receiver, hence we denote by Ω the set of receivers associated to the vertices in $\mathcal{G}$. If $|\mathcal{G}| < i$, i.e., if we were not able to select at least i vertices, then we return two empty sets (lines 19-20). Otherwise, we proceed by trying to select, for each receiver $\mu(v') \in \Omega$, $d_{\mu(v')} - 1$ additional lowest-cardinality vertices in $\mathcal{H}_i$ having no links with the vertices in $\mathcal{G}$ associated to receivers different from $\mu(v')$ (lines 10-17). If we succeed (or if the considered hierarchy is 1, meaning that we have no further hierarchies to explore), then we return the set of selected vertices $\mathcal{G}$ as well as the set of associated receivers Ω (line 18).

Example 2. *Consider the same conflict graph $\mathcal{H}_{\mathbf{M},\mathbf{W}}$ described in Example 1 and let us apply Algorithm 2. We start by constructing $\mathcal{K}_v$ by inspection of the conflict graph: $\mathcal{K}_{v_1} = \{1,2,3\}$, $\mathcal{K}_{v_2} = \{1,2\}$, $\mathcal{K}_{v_3} = \{1,2,3\}$, and $\mathcal{K}_{v_4} = \{1,2,3\}$. Hence, $\mathcal{H}_U = H_3 = \{v_1, v_3, v_4\}$, $\mathcal{H}_2 = \{v_2\}$, and $\mathcal{H}_1 = \emptyset$. Starting from the highest hierarchy, at line 5 of Algorithm 2 we call the subroutine GISfunction with parameters $\mathcal{H}_3$, v_1 and 3. By running iteratively the proposed algorithm, the GISs shown in Fig. 2 are obtained.*

V. Numerical Results

In this section, we provide numerical results for the performance of the proposed RAP-CA-HgC algorithm. We compare the RAP-CA-HgC algorithm with: *i)* LFU caching with Compound Channel transmission (i.e., naive multicasting at the rate of the worse channel receiver), referred to as LFU-CC; *ii)* LFU caching with unicast transmission, referred to as orthogonal LFU (O-LFU); *iii)* RAP caching with separate source-channel coding over compound channel with HgC as source encoder, referred to as RAP-SSC-CC; and *iv)* the benchmark upper bound (RAP-CA-HgC) when $D = \infty$ (see [1, Theorem 2]).

To illustrate the effectiveness of RAP-CA-HgC, we consider

[7]With an abuse of notation, we denoted with $c_u = \mathcal{C}_u(\{v' \in \mathcal{G} : \mu(v') = u\})$ the codeword obtained by coding the Bs_u source bits associated to the packets $\{\rho(v')\}$ with $v' \in \mathcal{G} : \mu(v') = u$.

Algorithm 2 CA-HgC$_2$

1: $\mathcal{A} = \emptyset$
2: $\mathbf{X} = \emptyset$
3: **for** $i = U : 1$ **do**
4:   **for** $v \in H_i : |\mathcal{K}_v| = \min_{v' \in H_i}\{|\mathcal{K}_{v'}|\}$ **do**
5:     $[\mathcal{G}, \Omega] =$`GISfunction`(H_i, v, i)
6:     **if** $\mathcal{G} \neq \emptyset$ **then**
7:       $\forall u \in \Omega$ code the vertices in $\{v' \in \mathcal{G} : \mu(v') = u\}$ with the $\Delta/\bar{\chi}$-length codeword $c_u = \mathcal{C}_u(\{v' \in \mathcal{G} : \mu(v') = u\})$ in the codebook $\mathcal{C}_u$
8:       $\mathbf{X} = [\mathbf{X}\ \sum_{u\in\Omega} \otimes\, c_u]$
9:       color $(\sum_{u\in\Omega} \otimes\, c_u)$ by $\alpha \notin \mathcal{A}$
10:       $H_i = H_i \setminus \mathcal{G}$
11:     **else**
12:       $H_i = H_i \setminus \{v\}$; $H_{i-1} = H_{i-1} \cup \{v\}$
13:     **end if**
14:   **end for**
15: **end for**
16: **return** $|\mathcal{A}|$, $\mathbf{X}$

Algorithm 3 `GISfunction`(H_i, v, i)

1: $\mathcal{G} = \{v\}$; $\Omega = \{\mu(v)\}$; $\tilde{H}_i = H_i$
2: **for** $v' \in \tilde{H}_i \setminus \mathcal{G} : |\mathcal{K}_{v'}| = \min_{v'' \in \tilde{H}_i \setminus \mathcal{G}}\{|\mathcal{K}_{v''}|\}$ **do**
3:   **if** no edge between v' and $\mathcal{G}$ **then**
4:     $\mathcal{G} = \mathcal{G} \cup \{v'\}$; $\Omega = \Omega \cup \{\mu(v')\}$
5:   **else**
6:     $\tilde{H}_i = \tilde{H}_i \setminus \{v'\}$
7:   **end if**
8: **end for**
9: **if** $|\mathcal{G}| \geq i$ **then**
10:   **for** $v' \in \tilde{H}_i \setminus \mathcal{G} : |\mathcal{K}_{v'}| = \min_{v'' \in \tilde{H}_i \setminus \mathcal{G}}\{|\mathcal{K}_{v''}|\} \wedge \mu(v') \in \Omega$ **do**
11:     $\mathcal{G}_{\mu(v')} = \{v" \in \mathcal{G} : \mu(v") = \mu(v')\}$
12:     **if** $|\mathcal{G}_{\mu(v')}| < d_{\mu(v')}$ and no edge between v' and $\mathcal{G} \setminus \mathcal{G}_{\mu(v')}$ **then**
13:       $\mathcal{G} = \mathcal{G} \cup \{v'\}$
14:     **else**
15:       $\tilde{H}_i = \tilde{H}_i \setminus \{v'\}$
16:     **end if**
17:   **end for**
18:   **return** $\mathcal{G}$, Ω
19: **else**
20:   **return** $\mathcal{G} = \emptyset$, $\Omega = \emptyset$
21: **end if**

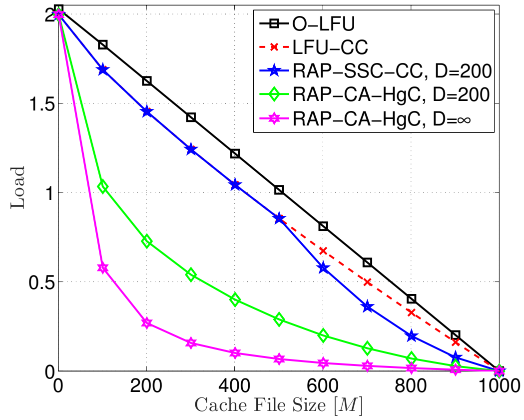

Fig. 3. Network load with $m = 1000, U = 30, \alpha = 0.2$.

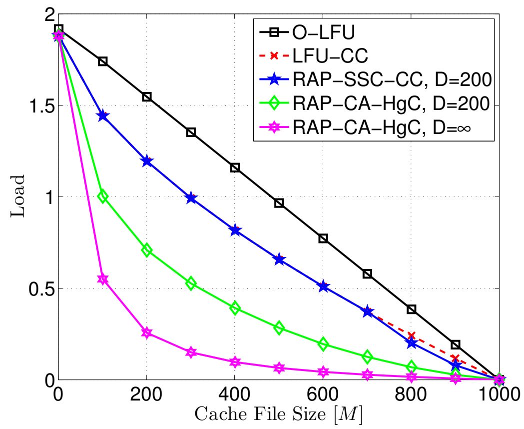

Fig. 4. Network load with $m = 1000, U = 30, \alpha = 0.4$.

a scenario where receivers request files according to a common Zipf demand distribution with $\alpha = \{0.2, 0.4\}$ and all caches have size M files. In line with [6], [7], we choose a uniform caching distribution ($p_f = 1/m, \forall f$), shown to be order-optimal for any $\alpha < 1$ in the error-free scenario. Moreover, we show the results in terms of average *network load*, defined as the inverse of the total multicast rate delivered by the sender over the wireless channel, and given by $\frac{\bar{X}}{\sum_{u=1}^{U} \eta_u}$, as shown in [1]. This metric is specially illustrative of the the amount of bandwidth resources the wireless operator needs to provide in order to meet the receiver demands. Fig. 3 shows the load for a network with $U = 30$ receivers, channel rates uniformly distributed among three values typically used in LTE standards: $\{\eta_1 = \frac{1}{2}, \eta_2 = \frac{3}{4}, \eta_3 = \frac{1}{4}\}$, maximum number of descriptions $D = 200$, and Zipf parameter $\alpha = 0.2$. We note that RAP-CA-HgC significantly outperforms all the state of the art schemes. Specifically, differently from RAP-SSC-CC, RAP-CA-HgC is able to preserve the promising multiplicative caching gain, even in the presence of receivers with different channel conditions and finite maximum number of descriptions D. The same considerations hold by varying the ZIP parameter α, as shown in Fig. 4.

VI. Conclusions

In this paper, we design and analyze a novel wireless video delivery scheme that specifically addresses the pressing need to accommodate next-generation video-based services over increasingly crowded and bandwidth-limited wireless access networks. Our solution considers a heterogeneous wireless edge architecture where the receivers are connected to the sender by lossy links. We provide an information theoretic formulation for the channel-aware caching-aided coded multicast problem and design two achievable schemes, RAP-CA-CIC and RAP-CA-HgC, which result from the careful implementation of joint source-channel coding to the caching-aided coded multicast problem. Our solutions guarantee the highest admissible video delivery rate to each receiver for the given propagation conditions, i.e., completely avoiding throughput penalizations from the presence of receivers experiencing worse propagation conditions.

References

[1] A. S. Cacciapuoti, M. Caleffi, M. Ji, J. Llorca, and A. M. Tulino, "Speeding up future video distribution via channel-aware caching-aided coded multicast," *IEEE J. Select. Areas Commun.*, 2016.

[2] C. V. N. Index, "Forecast and methodology, 2013-2018," Cisco, June 2014.

[3] ——, "Globe mobile data traffic forecast update, 2014-2019," Cisco, Febr. 2015.

[4] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *arXiv:1209.5807*, 2012.

[5] M. Maddah-Ali and U. Niesen, "Decentralized coded caching attains order-optimal memory-rate tradeoff," *IEEE/ACM Trans. on Networking*, vol. 23, no. 4, pp. 1029–1040, Aug. 2015.

[6] M. Ji, A. Tulino, J. Llorca, and G. Caire, "On the average performance of caching and coded multicasting with random demands," in *Proc. of Int. Symp. on Wireless Commun. Systems (ISWCS)*, Aug. 2014.

[7] ——, "Order-optimal rate of caching and coded multicasting with random demands," *arXiv:1502.03124*, 2015.

[8] ——, "Caching and coded multicasting: Multiple groupcast index coding," in *Prof. of IEEE Global Conf. on Signal and Information Processing (GlobalSIP)*, Dec. 2014, pp. 881–885.

[9] M. Ji, K. Shanmugam, G. Vettigli, J. Llorca, A. Tulino, and G. Caire, "An efficient multiple-groupcast coded multicasting scheme for finite fractional caching," in *Proc. of IEEE Int. Conf. on Commun. (ICC)*, June 2015, pp. 3801–3806.

[10] G. Vettigli, M. Ji, A. Tulino, J. Llorca, and P. Festa, "An efficient coded multicasting scheme preserving the multiplicative caching gain," in *Proc. of IEEE Conf. on Computer Commun. Workshops (INFOCOM WORKSHOPS)*, April 2015, pp. 251–256.

[11] K. Shanmugam, M. Ji, A. Tulino, J. Llorca, and A. Dimkakis, "Finite length analysis of caching-aided coded multicasting," in *Proc. of Allerton Conf. on Commun., Control, and Computing*, Sept. 2014, pp. 914–920.

[12] P. A. Chou, H. J. Wang, and V. N. Padmanabhan, "Layered multiple description coding," in *Proc. of Packet Video Workshop*, April 2003.

[13] S. Pudlewski, N. Cen, Z. GUan, and T. Melodia, "Video Transmission Over Lossy Wireless Networks: A Cross-Layer Perspective," *IEEE J. Select. Topics in Signal Processing*, vol. 9, no. 1, pp. 6–22, Febr. 2015.

[14] N. Golrezaei, A. Molisch, A. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Communications Magazine*, vol. 51, no. 4, pp. 142–149, April 2013.

[15] S. Kokalj-Filipovic, E. Soljanin, and Y. Gao, "Cliff effect suppression through multiple-descriptions with split personality," in *Proc. of IEEE Int. Symp. on Inform. Theory (ISIT)*, July 2011, pp. 948–952.

[16] Z. Bar-Yossef, Y. Birk, T. Jayram, and T. Kol, "Index coding with side information," *IEEE Trans. Information Theory*, vol. 57, no. 3, pp. 1479–1494, 2011.