WHERE TO SPEND THE BITS? EFFICIENCY OF SOURCE AND CHANNEL CODING IN MBMS

Kemal Ugur[1], Stephan Wenger[1], Dong Tian[2] and Moncef Gabbouj[2]
[1] Nokia Research Center, Tampere, Finland
[2] Tampere International Center for Signal Processing (TICSP), Tampere, Finland

ABSTRACT

In its Release 6, the Third Generation Partnership Project (3GPP) is defining a new service known as Multimedia Broadcast/Multicast (MBMS) that enables a number of new applications. Due to its nature, no feedback link from the receiver to the sender exists in MBMS. Hence no retransmission techniques can be employed to cope with the underlying erroneous wireless channel. Instead, 3GPP is adopting a channel coding technique based on a Forward Error Correction (FEC) scheme at the application layer. In this work, we are trying to find a good balance of source and channel coding to achieve the best video quality under MBMS conditions. We use a simulation environment that closely represents the channel behaviour of the 3GPP wireless link and compare cases with different FEC overheads at different error rates. Experiments show that careful selection of FEC overhead yields to significantly better video quality.

1. INTRODUCTION

Multimedia communication is one of the hot research topics in academia and industry. Recently, a lot of work in this wide field has focussed on the upcoming services offered by the third generation mobile networks.

The Third Generation Partnership Project (3GPP) is the key standardization body for third generation network environments. 3GPP's philosophy is to make use of, and to trigger (when necessary), standardization activities in bodies such as the IETF, ISO/IEC and ITU-T, wherever possible. The application layer protocols, for example, are almost exclusively developed in the IETF. Sometimes, however, 3GPP ventures into the development of its own specifications. This paper evaluates the performance and the best operation point of one of these mechanisms.

Optimized, joint source/channel coding is a research topic as old as signal processing. Shannon's famous separation theorem of 1948 [1] could be expressed in stating that source coding (compression) and channel coding (error protection) can be performed separately and sequentially, while maintaining optimality. Since 1948, it has been shown numerous times that the theorem holds only for idealistic environments; see for example Verdú's work [2]. Today, it seems consensus that source coding based error resilience is virtually irreplaceable, particularly for low latency applications.

Streaming applications are characterized in that they have relaxed real-time constraints when compared to conversational applications. The relaxed real-time constraints allow for the use of large buffers, which, in turn, should result in efficient channel coding. Hence, streaming may be an operation point where Shannon's theorem may start working.

3GPP, in its Release 6, defines two different streaming services: Point-to-Point Streaming Service (PSS) and Multimedia Broadcast/Multicast Service (MBMS). While the delay requirements of both services are similar, the available mechanisms in the transmission path are not: PSS, being a point-to-point service, allows for retransmission, whereas MBMS does not. In fact, it is assumed that almost all PSS traffic will be utilizing link layer Automatic Repeat Request (ARQ) mechanisms, so to guarantee an almost error free channel. For this reason, we focus on MBMS, where the multicast characteristic disallows the use of ARQ.

Later sections of this paper provide a detailed description of the protocol hierarchy employed by MBMS. For not it should suffice to say that an IP/UDP/RTP protocol hierarchy is used above the 3GPP MBMS bearer channel. These packets are subject to erasures. In order to allow for a high reproduced media quality, a forward error correction (FEC) framework has been developed in 3GPP, that operates on the RTP payload level and combats erasures. The FEC framework adapts gracefully to fluctuating packet sizes (where RFC 2733 [9] would require excessive padding), allows for the use of large source blocks (where RFC2733 allows a maximum of 32 packets in one source block) and facilitates the "plug in" of different FEC codes (where RFC2733 offers only a simple XOR mechanism). All these features warrant a closer look to this mechanism.

As for the media coding we choose the H.264 video compression technology. Without doubt, this video codec, when configured appropriately, offers the best compression efficiency of all standardized video codecs today. We choose the encoder settings such that only minimal source coding based error resilience was utilized. A few unconventional optimizations were used, as discussed in detail later, that allowed for some resilience against picture erasure, but no excessive intra coding, flexible macroblock ordering (FMO) or other expensive (in terms of source coding bit rate) tools have been used.

Using a simulation environment that closely represents the channel behaviour of a 3GPP wireless link, and the mentioned source and channel coding tools, we hope to find a good balance of source and channel coding.

The rest of this paper is organized as follows: section 2 reviews in the necessary detail the 3GPP MBMS environment, and section 3 very briefly introduces H.264 and the associated packetization, defined in RFC 3984 [3]. Section 4 discusses the simulation setup and the simulation results, respectively. An outlook in section 5 closes this paper.

2. OVERVIEW OF 3GPP MBMS

The 3GPP MBMS Technical Specification [4] defines a point-to-multipoint service in which multimedia data (audio, video, speech, text, still images etc.) is transmitted from a single source to multiple recipients, utilizing shared network resource. MBMS-based services use three distinct layers, Bearer, Delivery Method and Application, to deliver the multimedia data. These functional layers are depicted in Figure 1, and are described in the following subsections.

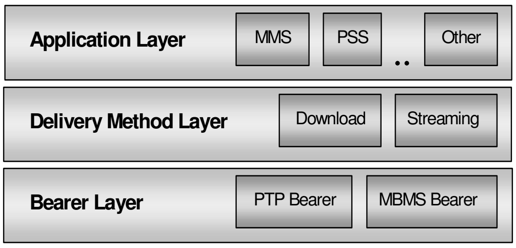


Figure 1 Functional Layers of MBMS

2.1 MBMS Bearer Layer

The MBMS Bearer provides the architecture to transport data from a single source to multiple recipients, and forms the basis of all MBMS services. It offers a receiver driven multicast operation, which implies the receivers join and leave multicast sessions, and the network forwards traffic only along paths that have downstream receivers [5]. The MBMS Bearer Architecture, as defined in [6], specifies two modes of operation, *broadcast mode* and *multicast mode*. Broadcast mode provides the mechanism to implement services similar to IP Broadcasting. In this mode, the tune-in to the broadcasted session is based exclusively on the user decision without the need of subscription. Thus the network is not aware of the number and characteristics of the tuned-in users. The multicast mode involves a subscription mechanism that establishes the relationship between the user and the service provider. In addition to some MBMS related functions at GGSN, SGSN, RNC/BSC, a functional entity is defined to support MBMS specific services. This functional entity is called Broadcast-Multicast Service Centre (BM-SC), and it provides centralized control of the MBMS user services. There are four functions defined by BM-SC which are essential for MBMS. These functions are *i) Membership*, *ii) Session and Transmission*, *iii) Proxy and Transport* and *iv) Service Announcement*.

The Bearer Layer provides *MBMS Bearer Service* to deliver IP multicast datagrams to multiple receivers. The interface of this service can be considered as an octet stream with units of data (PDU) of fixed size. The PDU integrity is achieved through a Layer-2 checksum, hence no bit errors can occur at the service interface, only erasures. Due to the point-to-multipoint architecture of MBMS, no feedback link can exist from the receivers to the sender. Therefore MBMS cannot rely on advanced retransmission strategies, such as ARQ, for error correction – all error control has to be implemented in the forward channel.

2.2 MBMS Delivery Method Layer

MBMS utilizes two different delivery methods, namely Download and Streaming, to deliver the multimedia content to the receiver. The protocol hierarchies used by the two different delivery methods are shown in Figure 2.

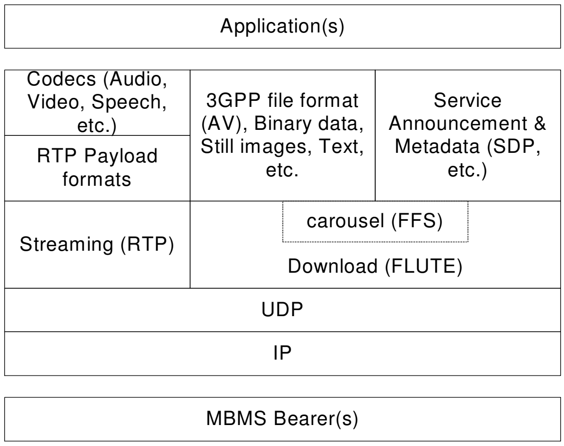


Figure 2 The MBMS Protocol Stack

The Download delivery method provides reliable multicast file download based on FLUTE protocol [7]. Optionally, a carousel mode may also be present and can be used for application such as text-TV. The streaming delivery method enables delivering continuous, close to real-time multimedia data over an MBMS bearer and it is based on the Real-time Transport Protocol (RTP). Further details of streaming delivery method is presented in Section 2.4.

2.3 MBMS Application Layer

The MBMS Application Layer is located hierarchically above the Bearer and Delivery Method layers, and enables different MBMS services and applications. Some examples of MBMS applications could be provided as follows:

a) *News Distribution:* The user registers to MBMS news broadcast service. News clips are distributed to the users in the form of text, still picture or low-quality audio/video.
b) *Streaming Audio:* The musical entertainment sessions, like “Top-10 hits of the week”, are streamed to the users
c) *Localized Services:* Local tourist information is continuously broadcasted, and the user tunes-in to learn the essentials of the city.
d) *Software Update:* If the operator needs the update the software on a certain terminal type, the update is sent to those terminals over a multicast channel.

2.4 Streaming in MBMS

MBMS uses RTP/UDP/IP protocol hierarchy, to stream continuous media data to the users. The data, generated by a specific media codec, is first packetized to RTP packets. RTP packets consist of at least 12 bytes of RTP header and a payload of variable size, carrying the payload header and the media bits. The RTP packets are encapsulated in UDP/IP packets. The size of the UDP header is 8 bytes and the size of the IP header is 20 bytes, resulting in a total of 28 bytes of overhead for encapsulating RTP to IP packets. The IP packets are framed, using a protocol such as PPP, and mapped to the Layer-2 PDUs of the MBMS Bearer service, which interfaces with the physical layers for transmission. The encapsulation of media data in RTP/UDP/IP packets and the mapping of IP packets to PDUs are depicted in **Figure 3**.

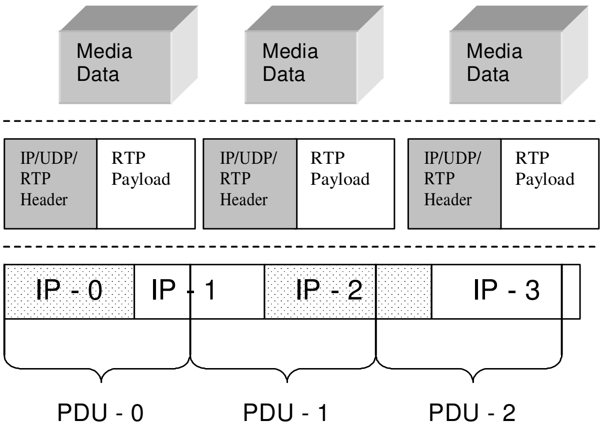


Figure 3 MBMS IP Protocol Hierarchy

Due to nature of compressed media, the IP packets are usually at variable sizes. The mapping of variable sized IP packets to fixed size PDUs is performed by concatenating IP packets (and the framing information) back-to-back. Therefore, the boundaries of IP packets are not aligned with the layer 2 PDUs. Because of this, the loss of one PDU at Layer-2 corresponds to loss of one or more IP packets, depending on the size and position of IP packets. For example, if PDU-0 in **Figure 3** is lost, IP Packets 0 and 1 are lost. If PDU-1 is lost, IP Packets 1 and 2 are lost, and so forth.

3. H.264 VIDEO CODING STANDARD

H.264 is the newest video coding standard, jointly developed by MPEG and ITU committees, and it achieves 50% bitrate savings over previous standards. It includes wide range of tools making it suitable for different applications with different requirements. In addition to significantly improved compression efficiency, H.264 is also designed to have improved network adaptation. It uses a new concept of Video Coding Layer (VCL) and Network Adaptation Layer (NAL) to increase the network friendliness of the system. The coding of video signal is specified at VCL, whereas NAL specifies the interface between the video codec and the underlying network that transports video data. H.264 also introduced *parameter set* concept that allows information that is staying constant for the duration of at least one picture to be transmitted out-of-band. More specifically, *sequence parameter sets* are comprise information related to an IDR picture and the subsequent picture until the next IDR picture, whereas *picture parameter sets* contain information related to all slices within a picture. H.264 includes several error resilience tools that improve its performance over erroneous channels.

Using non-reference frames is a common technique to increase the error-resiliency of the system, as no temporal error-propagation can occur when a non-reference picture is partially or completely lost. A common example for non-reference pictures are MPEG-2's B frames. In H.264, due to the presence of long-term memory prediction and the reference picture reordering tool, all picture types can serve as non-reference pictures. We use in the following simulations two types of predictively coded frames; a P frame (capital letter) which carries forward prediction information, and a p frame (lower case letter) which is not used for future prediction. The notation IppP implies that two non-reference p pictures are placed between an I- and a P picture. Reference picture reordering allows us to keep the prediction distance short. The first p picture is predicted from the I picture, whereas the second p picture is predicted from the following P picture, without per macroblock signalling in the bit stream (the reference picture reordering commands MMCO are part of the slice header). The relatively inefficient prediction between the I and the P picture, due to the large temporal distance between the two pictures can be (at least partly) partly compensated by choosing a coarser QP for the p pictures.

Unlike previous standards, intra-coded macroblocks can be predicted from the neighbouring inter-coded macroblocks. This has a serious impact on error resilience, as error propagation is possible between corrupted inter-coded regions and the intra information. The constrained_intra_pred_flag bit in the picture parameter set disallows this form of prediction and is enabled for all simulations.

4. SIMULATION SETUP AND RESULTS

A simulation tool that closely represents the behaviour of a wireless link is used. We first encode the H.264 video at a rate so that video rate + FEC rate is equal and it is 128 kbps. We use the IppP structure, and packetize it as one NAL unit per RTP packet. This type of packetization is called "Single NAL Unit" mode of RFC 3984 [3], and it does not include advanced error-resilience techniques available in other modes indicated by the RFC. The maximum slice size of the pictures is adjusted according to the underlying PDU size in order to avoid fragmentation of the IP packets at Layer-2. We then assume FEC framing operates on 5 seconds of video, which means for a 15 fps sequence, one FEC frame contains 75 video packets (due to the large SDU sizes used in our simulations, one slice per frame was used for all the video frames), plus the FEC repair packets. We use an IDR picture as the first picture of the FEC frame, corresponding to an IDR frequency of 75 frames.

Our RTP packets contain the video + FEC data as the input for the MBMS channel simulator. Simulator assumes fixed PDU sizes having 80 ms. in duration. The PDU loss patterns provided at [8] for different error rates are mapped to lost IP and RTP packets. At the receiver side, we simulate FEC repair mechanism assuming that the lost video packets within a FEC frame can be recovered if the number of correctly received FEC repair packets is not less than the number of lost video packets. Otherwise, FEC fails to repair any of the packets resulting severe losses. The video packets are then extracted from the FEC frame, depacketized and decoded by the H.264 decoder.

We simulate 0%, 0.5%, 1.0% and 1.5% PDU loss rates at Layer-2 and we simulate a FEC overhead of 0%, 5%, 10% and 15%. For more reliable simulation results, we manually increased the length of the source video to 7500 frames that corresponds to 500 seconds or 100 FEC frames, by concatenating the entire video file several times.

Figure 4 presents the decoded average PSNR for *Foreman* and *Paris* sequences at the aforementioned conditions. It is clearly seen that using the right balance of FEC and source coding, significantly affects the performance of the system. It is also seen that when FEC is not used, or used with a low percentage, the performance of the system drops significantly at high error rates. It should be noted that the average PSNR does not indicate the subjective quality of the decoded video, and even a small drop in PSNR at high error rates has severe impacts in visual quality. This is because the drop in PSNR is due to FEC frames that couldn't be repaired, because of too many lost packets contained within the FEC frame This results in a highly erroneous video segments lasting for the duration of the FEC (which is 5 seconds in our simulation). Due to the heavy errors in the video segment, error concealment techniques cannot work efficiently, and the decoded video has very poor subjective quality for a long period of time (in our experiments the PSNR for those segments are found to be around 20 dB).

5. CONCLUSIONS AND FUTURE WORK

This work, intends to study the MBMS framework from the video coding and transport perspective. More specifically, we have studied the effect of FEC in MBMS scenarios and try to come up with a good balance of source and channel coding. It is found out that, using more bits for FEC decreases the video quality at low error rates, but keeps the video quality significantly higher than the case of using less number of bits for the FEC. The decrease in subjective quality is very severe for the latter when the error rate starts to increase.

For a more accurate study, the actual FEC algorithms should be implemented and tested for the simulated channel. Also, it would be interesting to test the advanced error resilience features of H.264 in MBMS environments.

REFERENCES

[1] C. E. Shannon, "*A mathematical theory of communication*," Bell System Technical Journal, vol.27, pp.379-423, pp. 623-656, 1948

[2] S. Vembu, S. Verdú, and Y. Steinberg "*The Source-Channel Separation Theorem Revisited*," IEEE Trans. on Information Theory, vol. 41, no. 1, pp. 44-54, Jan. 1995.

[3] S. Wenger, T. Stockhammer, M.M. Hannuksela, M. Westerlund, and D. Singer, "RTP payload format for H.264 video," RFC 3984, February 2005

[4] 3GPP TS 26.346, *Technical Specification Group Servicesand System Aspects; Multimedia Broadcast/Multicast Service; Protocols and Codec.*

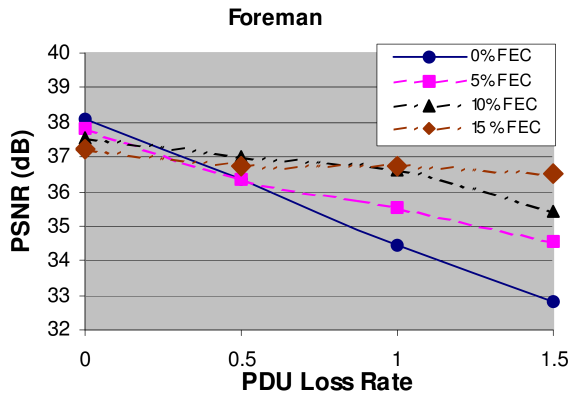
[5] S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven layered multicast," in Proc. SIGCOMM'96, ACM, Stanford, CA, Aug. 1996,pp. 117–130.

[6] 3GPP TS 23.246: "*MBMS Architecture and Functional description (Stage 2)*".

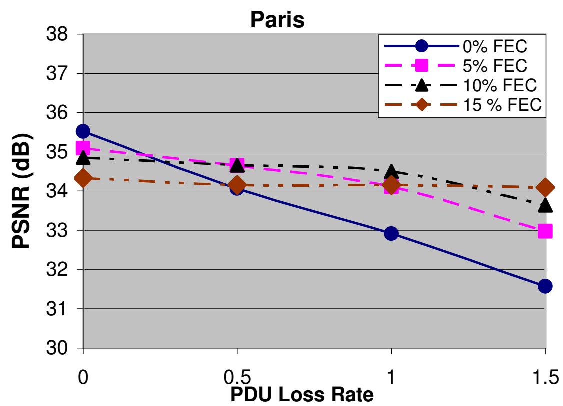
[7] T. Paila, M. Luby, R. Lehtonen, V. Roca, and R. Walsh, FLUTE - *File Delivery over Unidirectional Transport*, Internet-Draft, Jun. 2004.

[8] Doc. S4-040803, "*Video Network Simulator and Error Masks for 3GPP Services*", found in ftp.3gpp.org

[9] J. Rosenberg and H. Schulzrinne, "An RTP Payload Format for Generic Forward Error Correction", RFC 2733, December 1999



(a)



(b)

Figure 4 Simulation Results for (a) Foreman (b) Paris