ASYNCHRONOUS MEDIA STREAMING OVER WIRELESS BROADCAST CHANNELS

Hrvoje Jenkač

Institute for Communications Engineering
Munich University of Technology, Germany
h@tum.de

Thomas Stockhammer

Siemens AG - Mobile Phones
Munich, Germany
stockhammer@nomor.de

ABSTRACT

We consider wireless broadcasting of multimedia content to allow asynchronous media access. Receivers subscribe at any time to the ongoing broadcast session, but are still able to display the media stream from the beginning. A fully scalable broadcasting scheme is presented where the media stream is appropriately segmented and segments are protected by fountain codes. The decoding behavior of rateless codes on wireless as well as on erasure channels is considered within the framework of *information collection*. Asynchronous data access and full reliability at the same time are achieved. Depending on its receiving conditions the receiver adapts its initial playout delay for high probability of successful playout. Analytical expressions for the failure probability of successful media playout are derived depending on the initial delay and the channel conditions at the receiver.

1. INTRODUCTION

In traditional broadcast environments all receivers retrieve an identical media stream synchronously. Though late tune-in is commonly supported by random access points in the bit-stream, a delayed start of data reception results in the miss of a significant amount of earlier content, depending on the access time. In the contrast, video-on-demand schemes allow media requests at arbitrary times. Trivial schemes operating with individual point-to-point connections require an infeasible amount of transmission bandwidth with increasing number of receivers. Bandwidth in general is a costly resource, especially in wireless environments. A huge variety of broadcasting schemes for this purpose has been developed and investigated [1, 2]. All of them have in common that media can be requested asynchronously by receivers, with tolerable and fixed bandwidth expansion, independent of the number of receivers, as well as acceptable waiting times. However, these schemes only solve the problem of asynchronous media access, but do not provide any loss resilience. Unfortunately, packet networks such as the Internet suffer from losses. Therefore, the proposed broadcast schemes have been extended to operate in lossy environments [3]. Additional error correction has been introduced based on fixed rate channel codes, resulting in additional bandwidth requirement. Recently in [4] it was outlined that fixed rate codes should be replaced by *rateless* codes or *fountain* codes. We describe a realization of the proposed scheme, and we extend this work to wireless broadcast, where additional disturbances due to fading, interference, and noise are experienced. We discuss the question, how long a receiver with specific channel conditions should delay the start of the media presentation in order to guarantee a certain presentation reliability.

2. PRELIMINARIES

2.1. Problem Formalization

Assume an encoded media-stream $\boldsymbol{\mathcal{P}} = \mathcal{P}_1, \ldots, \mathcal{P}_N$, containing media units $\mathcal{P}_n$ of size $|\mathcal{P}_n|$ and an assigned decoding time-stamp (DTS) $t_{\text{DTS}}(n)$, where $t_{\text{DTS}}(1) = 0$. The duration of the media sequence is defined as $T = t_{\text{DTS}}(N)$. Assume decoding of $\mathcal{P}_1$ starts at time $t_{d,1}$. Hence, the latest decoding time for all remaining data units is fixed as $t_{d,n} = t_{\text{DTS}}(n) + t_{d,1}$. If $\mathcal{P}_n$ is not available at the receiver at time $t_{d,n}$ it is considered as lost. Due to predictive coding, e.g., in video, the loss of single data units might have significant impact on the quality. Therefore, to provide sufficient quality to the end user we define as *successful playout* only those events for which all data units can be decoded before their $t_{d,n}$. Obviously, this is a very restrictive approach and in practical systems some means to also decode non-successfully received multimedia sequences might be acceptable at the expense of degraded quality [5]. This is not subject of this work. With this abstraction for the multimedia application a receiver has basically only a single parameter to adjust, namely the specification of the decoding time for the first data unit, $t_{d,1}$.

In unicast streaming environments, the receiver requests a certain multimedia clip. The streaming server starts transmitting data unit $\mathcal{P}_1$ and with some processing and algorithmic delay, the decoder starts receiving $\mathcal{P}_1$ at time t_r.

0-7803-9332-5/05/$20.00 ©2005 IEEE

The initial playout delay between the multimedia request and the playout of the first data unit, Δ, is determined as $\Delta = t_{d,1} - t_r$. It is obvious that there exists a tradeoff in the selection of the initial delay Δ: Large Δs guarantee that it is likely that all data units are received in time, but they are annoying to the end user. In [5], we have developed a framework for unicast streaming which allows to find upper bounds on the probability that successful playout is not possible for different initial playout delays depending on the statistics of a wireless channel.

In a broadcast environment, the scenario is different. The media server broadcasts content without receiving a request from users. Unfortunately, users can only synchronously join broadcast sessions by tuning in at a *pre-determined* receiving time $t_{r,1}$ and playout is started at a *predetermined* decoding time $t_{d,1}$. In general it is not possible for users in a broadcast scenario to request individual transmission of a stream due to missing feedback links as well as due to missing capacities in the downlink to support the request of each user individually. So the following questions arise: (i) How can low-delay and efficient asynchronous media delivery be achieved in lossless broadcast environments? (ii) How can reliability be achieved for asynchronous wireless multimedia transmission? (iii) Can these concepts be combined to obtain sufficient reliability, low delay, and efficient channel exploitation? (iv) What are the tradeoffs between initial playout delay and successful playout probability depending on the receiving conditions for individual users participating in asynchronous wireless multimedia broadcast?

2.2. Asynchronous Media Delivery

To formalize the concept of asynchronous delivery in broadcast environments, we assume that the receiver again "requests" a media stream at some arbitrary time t_r by tuning into an ongoing broadcast rather than issuing a request to the streaming server. An obvious solution for asynchronous reception is repetition of the stream, referred to as *broadcast disk*, there the waiting time can be as large as $\Delta = T$. Reduction of the waiting time can be achieved by expanding the bandwidth by some factor ω. A trivial approach would be to broadcast the same media stream on K parallel channels. On each channel a delayed version of the media stream is offered with a delay shift of T/K. This allows receivers to successfully playout the stream with a worst case delay of $\Delta = T/K$ resulting in a bandwidth expansion of $\omega = K$. More sophisticated schemes [1, 2] have been proposed which allow to reduce the bandwidth expansion without increasing the initial delay. Most of them have in common that the media bitstream is segmented into segments $\mathcal{S}_i$ whereby each segment is periodically transmitted on different channels, but each channel is assigned a different transmission rate. The segments $\mathcal{S}_i, i = 1, \ldots, S$ are obtained by segmenting the entire media stream into S segments, such that each segment contains a number N_i consecutive data units $\mathcal{P}_n$. Furthermore, we basically treat each segment as super data unit such that each segment gets assigned a segment size $|\mathcal{S}_i|$ as well as a latest decoding time $t_{\text{DTS}}(\mathcal{S}_i)$ representing the earliest decoding time of any contained data units in this segment.

3. FEC FOR ASYNCHRONOUS RECEPTION

3.1. FEC for Erasure Channels

Assume a single segment $\mathcal{S}$ composed of k source packets of equal size being transmitted periodically. In case of unreliable transmission packets are lost. Either one has to deal with errors in the decoded media, or has to wait even much longer before this exact replica of the lost data is transmitted some time later. To increase reliability, a Forward Error Correction (FEC) code with some fixed code rate r to this single data stream could be applied. Moreover, in in a broadcast environment different receivers experience different loss rates and different loss patterns. Assuming that receivers tune in at arbitrary time, receivers with error-free channels have to accept that they receive a significant amount of useless information, namely the parity packets. In contrast, for bad users, the parity packets might not be sufficient to decode all original source packets such that they have to wait for another round in the broadcast disk. Again, these receivers generally receive a significant amount of useless data.

To avoid these problems, the notion of *rateless* channel codes has been coined indicating that an infinite amount of parity packets n for k source packets can be generated, *ie*, the code rate approaches 0. These codes are also referred to as *fountain codes*. In the following we outline the basic concept of rateless codes and their application for asynchronous and reliable data broadcast. For a detailed description we refer to [6, 7]. An potentially infinite number of encoding packets n from k source packets is generated on the fly, which are broadcasted consecutively. An ideal fountain code (on an erasure channel) is defined having the property to reconstruct the entire source message reliably from any k received encoding packets. This allows receivers to subscribe to the ongoing broadcast session at arbitrary time and to collect any k packets not even in a consecutive manner. In lossy environments receivers just wait until they receive any k packets before being able to start playout. No receiver receives useless information.

Although ideal fountain codes have not been found so far, practical codes are capable to approach the performance of ideal codes very well. Usually only a slightly increased number k' of packets have to be received. Practical approximations of a *digital fountain* for packet loss channels have been proposed by LT-Codes and Raptor codes [6] approaching $\epsilon \triangleq (k'-k)/k \to 0$.

3.2. Generalized Framework: Information Collection

Consider a single segment $\mathcal{S}$ containing $k = |\mathcal{S}|$ bits. $\mathcal{S}$ is encoded with a fountain code $\mathcal{F}$ to obtain a code word $\mathcal{X} = \mathcal{F}(\mathcal{S})$ of length $\tilde{n}$ whereby $\tilde{n} \to \infty$. Assume that code word $\mathcal{X}$ is sequentially broadcasted block-wise over radio slots m containing M bits each resulting in $\hat{n} = \tilde{n}/M \to \infty$ transmission slots. Within each slot m, a certain receiver observes a channel state γ_m. Assume that γ_m is a realization of a random variable Γ_m, e.g., fading in wireless transmission. Let us define a receiver pattern $\boldsymbol{r} = (r_1, r_2, \ldots)$, with $r_m \in \{0, 1\}$ and $r_m = 1$ indicating slots m to which the receiver listens to. Let $p_o(k, \boldsymbol{r})$ denote the probability that the information of size k cannot be decoded when listening to radio slots according to a specific receiver pattern $\boldsymbol{r}$. In practice $p_o(k, \boldsymbol{r})$ for a certain fountain code in combination with a specific decoder can be obtained by appropriate simulation campaigns. In oder to evaluate information theoretic limits of fountain codes operating on other channel types than the erasure channel, we defined in [8] *asymptotically optimal fountain codes* to fulfill the following property:

$$p_o(k, \boldsymbol{r}) = \Pr\left\{ M \sum_{m=1}^{\infty} r_m \cdot R_I(\Gamma_m) \leq k \right\}, \qquad (1)$$

with $R_I(\gamma)$ the mutual information between the input and the output for this specific channel state γ. This framework generalizes the idea of asynchronous reception which has almost exclusively been considered for erasure channels up to now. From a receiver point of view, it basically collects information from slots with certain channel states γ_m until sufficient information is available for successful decoding the specific fountain. This receiver operation is denoted as *asynchronous information collection*. Moreover, in [8] we showed that LT-codes in combination with soft-in belief propagation decoding operate quite close to the optimum in (1) when operated on the AWGN and fading channel.

4. ASYNCHRONOUS WIRELESS STREAMING

4.1. Rateless Broadcasting for Asynchronous Reception

Consider a segmentation $\mathcal{S}_i$ as introduced in Section 2.2 by any appropriate algorithm [1, 2]. The optimization of the segmentation is not considered in this work. Then, each segment $\mathcal{S}_i$ containing $|\mathcal{S}_i|$ information bits is encoded with an individual fountain code $\mathcal{F}_i$, *ie*, $\mathcal{X}_i = \mathcal{F}_i(\mathcal{S}_i)$. Slotted transmission of each segment fountain code word, $\mathcal{X}_i$, is performed, whereby each segment fountain $\mathcal{X}_i$ gets assigned an individual transmission rate Ω_i. We assume that the overall transmission rate Ω is shared among different segments by any orthogonal multiplexing method such that $\sum_i \Omega_i = \Omega$. Therefore, segment fountains $\mathcal{X}_i$ are alternatingly mapped to transmission slots with weight according to their rate Ω_i. The multiplexing period α determines the number of transmission slots such that within one period a certain segment gets assigned $\alpha\Omega_i/\Omega$ transmission slots. At the receiver it is not important which radio slots have been received, but only that sufficient information for each fountain is collected for successful decoding. Hence, this scheme provides two features at the same time: (i) full reliability can be achieved if the receiver just waits long enough until sufficient information is collected; (ii) asynchronous reception is realized by the properties of fountain codes. The playout delay for each user can be adapted to its actual channel characteristics.

4.2. Service Outage

In order to allow proper presentation of the media at the receiver the reconstruction of each segment $\mathcal{S}_i$ should take place before its reconstruction deadline, $t_{\text{RTS}}(\mathcal{S}_i)$, expires. However, the receiver obviously wants to start decoding and presenting the first segment $\mathcal{S}_i$ although the reception for the remaining segments is still ongoing. Hence we are interested how long a receiver should delay the presentation of the first segment in a lossy environment in order to achieve a desired service reliability? To answer this question, we assume that the receiver is at least aware of its channel statistics Γ_m, e.g., the loss probability of radio blocks or the distribution of the SNR, and that the channel is stationary during the reception time of the sequence. Without loss of generality, we assume that the receiver tunes into the ongoing broadcasted session at $t_r = 0$. Assume that the decoder decides for a playout delay Δ: Then, with segment DTS $t_{\text{DTS}}(\mathcal{S}_i)$, the latest time each segment fountain $\mathcal{X}_i$ must be reconstructed to avoid a playout problem results in $t_{\text{RTS}}(\mathcal{S}_i) = t_{\text{DTS}}(\mathcal{S}_i) + \Delta$. Let $\mathcal{N}_i(t)$ denote the number of transmitted radio slots from segment fountain $\mathcal{X}_i$ up to time t. We assume that once the receiver subscribes to the broadcast session it stays tuned and, therefore, the receiver likewise listens to $\mathcal{N}_i(t)$ radio blocks up to time t resulting in an equivalent receiver pattern $\boldsymbol{r}_i(t)$ for segment fountain $\mathcal{X}_i$, with $r_{i,m} = 1\,\forall_{1\leq m\leq \mathcal{N}_i(t)}$ and $r_{i,m} = 0\,\forall_{m>\mathcal{N}_i(t)}$. Then, the probability $p_{\text{out}}(\mathcal{S}_i)$ that segment $\mathcal{S}_i$ cannot be reconstructed in time results in $p_{\text{out}}(\mathcal{S}_i) = p_o(|\mathcal{S}_i|, \boldsymbol{r}_i(t_{\text{RTS}}(\mathcal{S}_i) = t_{\text{DTS}}(\mathcal{S}_i) + \Delta))$. Therefore, for i.i.d channel states Γ_m, the probability $P_{\text{OUT}}(\Delta)$ that at least one segments cannot be reconstructed in time, and, hence the media sequence cannot be played out successfully yields

$$P_{\text{OUT}}(\Delta) = 1 - \prod_{i=1}^{S} \Big(1 - p_o\big(|\mathcal{S}_i|, \boldsymbol{r}_i(t_{\text{DTS}}(\mathcal{S}_i) + \Delta)\big)\Big). \quad (2)$$

Assuming that the receiver is aware of the channel statistics Γ_m, the media segmentation $\mathcal{S}_i$, the segment decoding time stamps $t_{\text{DTS}}(\mathcal{S}_i)$, and the transmission rates Ω_i, each receiver can decide how long to wait in order to achieve a sufficiently small service outage probability $P_{\text{OUT}}(\Delta)$. Alternatively, the calculation can be performed at the transmitter in advance for a set of channel parameters [5] and is made available within the setup of the broadcast session.

5. PERFORMANCE EVALUATION

In order to validate our analytical findings, we evaluate (2) for *Harmonic Broadcasting (HB)* [2]. We consider a CBR media stream with data rate $R_d = 128$ kbit/s and duration $T = 2$ hours. HB was shown to be bandwidth-optimal for CBR media [2]. The bit-stream is segmented in segments $\mathcal{S}$ of equal size, *ie*, $\forall_i |\mathcal{S}_i| = k$. We select $k = \frac{T \cdot R_d}{S}$ for both $S \in \{18, 36\}$. The radio block size is selected as $M = 160$ bits, the multiplexing period α and Ω_i as $\alpha = (|\mathcal{S}| \cdot H(S))/M$ and $\Omega_i = 1/(i \cdot H(S))$ respectively, with $H(n) = \sum_{i=1}^{n} 1/i$. This results in a bandwidth expansion of $\omega = 3.5$ and $\omega = 4.2$ for $S = 18$ and $S = 36$, respectively. The transmission time interval interval t_{tti} between two consecutive radio blocks is set to $t_{\text{tti}} = R_d \cdot H(S)/M$. We consider transmission over the AWGN channel for different E_S/N_0. Two modes of operation are compared. The first mode investigates the performance when erasure based decoding of the segment fountains is considered. For this, appropriate erasure declaration of radio blocks is required, *ie*, radio blocks are declared as erased at the receiver if residual bit errors within a radio block are detected. In the second scheme, erroneous radio blocks are not declared as erased, but soft information about each bit is propagated to the fountain decoder. For a more detailed description we refer to [8]. Our results were obtained with a combination of analytical and numerical methods, based on the assumption of asymptotically optimal fountain codes with properties according to (1). We evaluated (2) for a different E_S/N_0 and determined the required initial playout delay Δ such that the resulting outage probability is arbitrary small, *ie*, $P_{\text{OUT}} < 10^{-6}$. Figure 1 shows the required initial delay Δ over E_S/N_0 in dB. Basically, the following conclusions can be drawn. First, with decreasing E_S/N_0 the required initial delay is increasing. This comes from the fact that more time is required in order to allow successful reconstruction of every segment. Second, with increasing bandwidth expansion, *ie*, with increasing number of segments, the initial delay can be reduced. Soft decoding outperforms the erasure based scheme and reduces the required Δ significantly. Finally, even if no losses are present (for high E_S/N_0) a minimum initial delay is required resulting from the broadcasting scheme itself.

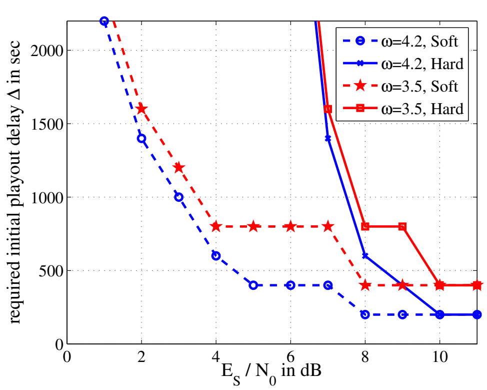

Fig. 1. Performance results: Initial delay Δ vs. E_S/N_0.

6. CONCLUSIONS

We considered wireless broadcasting of multimedia content with asynchronous media access. Receivers subscribe at any time to the ongoing broadcast session, but are still able to display the media stream from the beginning. We followed the ideas in [4] and extended the work to wireless environments. Thereby, the media stream is segmented and each segment is protected by a rateless codes. Each encoded segment is broadcasted at different rate. We introduced means for evaluating the decoding behavior of rateless codes on wireless as well as on erasure channels. The presented framework allows asynchronous data reception as well as full reliability at the same time. Further, error control is fully shifted to the receiver side and is achieved by delayed media presentation. The presented scheme is fully scalable and allows an arbitrary number of heterogeneous receivers. Finally, we derived analytical expressions for the probability of non-successful media playout. This allows receivers to adjust a desired service reliability by delaying the playout for a sufficient time. Performance results were presented showing the practicality of the discussed framework. Future work will consider scalable media and the overhead of practical fountain codes.

7. REFERENCES

[1] A. Hu, "Video-on-Demand Broadcasting Protocols: A Comprehensive Study," in *Proc. IEEE Infocom 2001*, Anchorage, Alaska, Apr. 2001.

[2] L. Engebretsen and M. Sudan, "Harmonic Broadcasting is Bandwidth-Optimal Assuming Constant Bit Rate," in *Proc. Annual ACM-SIAM Symposium on Discrete Algorithms 2002*, San Francisco, CA USA, Jan. 2002.

[3] L. Xu, "Efficient and scalable on-demand data streaming using uep codes," in *Proc. ACM International Conference on Multimedia*, Ottawa, Ontario, Canada, Sept. 2001.

[4] C. Huang, R. Janakiraman, and L. Xu, "Optimal loss-resilient media streaming using priority encoding," in *Proc. Intern. Conf. on Multimedia (ACM), New York, USA*, Oct. 2004.

[5] T. Stockhammer, H. Jenkac, and G. Kuhn, "Streaming video over variable bit-rate wireless channels," *IEEE Trans. Multimedia*, vol. 6, pp. 268–277, Apr. 2004.

[6] J. Byers, M. Luby, and M. Mitzenmacher, "A digital fountain approach to asynchronous reliable multicast," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 8, pp. 1528–1540, Oct. 2002.

[7] M. Mitzenmacher, "Digital fountains: A survey and look forward," *Proc. of the IEEE Information Theory Workshop 2004, San Antonio, TX, USA*, pp. 271–276, Oct. 2004.

[8] H. Jenkac, T. Mayer, T. Stockhammer, and W. Xu, "Soft Decoding of LT-Codes for Wireless Broadcast," in *Proc. IST Mobile Summit 2005*, Dresden, Germany, June 2005.