Video Streaming using Content-Aware Unequal Error Protection Fountain Codes

Michael Schier
University of Innsbruck, Austria
michael.schier@uibk.ac.at

Michael Welzl
University of Oslo, Norway
michawe@ifi.uio.no

***Abstract*—We present a novel mechanism for estimating the perceptual importance of network packets constituting a video stream. We explain how this information can be used to extend existing rateless coding schemes and present first results demonstrating the expected increase in perceptual quality.**

I. Introduction

Digital Fountain codes are a new class of error-control codes which are characterized by the ability to create an infinite number of encoded symbols. In this paper, we focus on one instance of this class, LT codes, invented by Luby [1]. By adapting the probability distribution by which input symbols are chosen to contribute to the calculation of encoded symbols, we are able to increase the probability of correctly reconstructing perceptually more important packets at the receiver. The perceptual relevance estimation process of single portions of the video stream incorporates the analysis of the fundamental building blocks of MPEG-like video codecs. We already demonstrated the effectiveness of an earlier version in [2]. This paper presents an extended version of the estimation algorithm which incorporates inter frame motion information by considering motion vectors and is tailored to the specific video codec H.264/MPEG-4 AVC.

The idea of manipulating probabilities at the rateless decoder to perform unequal error protection was first proposed by Rahnavard et al. [3]. To the best of our knowledge, there exists no related work which explores the properties of video streams to optimally exploit this feature, except Talari et al. [4] which merely considers frame types of MPEG2 streams as decision criterion.

II. Perceptual Relevance Estimation Mechanism

The estimation process basically takes three major factors into account: the types of single macroblocks and their spatial complexity, the number of temporally dependent macroblocks and the spatial distance and relevance of regions that they temporally refer to. We consider the perceptual relevance of a specific frame region as the mean over all estimates $\Psi(m_i)$ of macroblocks m_i it contains. Ψ is the weighted sum of all three factors mentioned before and is defined as $\Psi(m) = \sum_j \omega_j \cdot \Psi_j(m) \in [0;1]$ with $j \in \{\text{type, dep, mv}\}$ and $\sum_j \omega_j = 1$.

For the calculation of $\Psi_{\text{type}}(m)$, a lookup table is used which is the result of an iterative tuning process and numerous simulations. It encompasses 18 entries and contains groups of combinations of [sub-]macroblock types as keys, which are defined in tables 7.11-7.18 of the H.264 standard [5]. Roughly formulated, Ψ_{type} assigns rather high estimates to macroblocks of high spatial complexity (i.e. macroblocks with a high number of partitions/sub-partitions or intra-coded PCM macroblocks) whereas to macroblocks with no partitioning, low estimates are assigned. Furthermore, Ψ_{type} of SKIP-macroblocks is equal to zero.

$\Psi_{\text{dep}}(m)$ reflects the importance of macroblock m for temporally dependent macroblocks. We define $\delta_{\leftarrow}(m)$ to be the number of (distinctly) dependent elements which can be derived by inspecting the motion vectors of m and caching their targets to get the relations with inverted direction. This variable given, we define $\Psi_{\text{dep}} = 1 - \left(\frac{1}{\delta_{\leftarrow}(m)+1}\right)^{\kappa} \in [0;1]$ with $\kappa > 0$. The function Ψ_{dep} is strictly monotonically increasing, referable to the fact that a loss of macroblock m causes quality distortions in all dependent macroblocks. The factor κ is used to adapt Ψ_{dep} to the expected amount of temporal prediction, depending on the GOP structure used and the maximum number of reference frames.

Finally, $\Psi_{\text{mv}}(m)$ incorporates the distance between macroblock m and macroblocks of other frames used by m for temporal prediction. Broken temporal dependencies lead to distortion which decoders try to mitigate by interpolating the lost information. The higher the spatial and temporal distances between m and its references, the less likely it is to obtain an acceptable interpolation. As a consequence, scenes with high motion activity (which contain many macroblocks having motion vectors with lengths above average) are more vulnerable to data loss than slow-motion sequences with respect to decoded video quality. Based on these considerations, we define Ψ_{mv} as follows:

$$\Psi_{\text{mv}}(m) = \sum_{v_i \in MV(m)} \frac{\omega_{\text{REF}} \cdot \frac{len(v_i)}{\sqrt{w_f^2+h_f^2}} + (1-\omega_{\text{REF}}) \cdot \frac{|\Delta_{\text{REF}}(v_i)|}{max\Delta_{\text{REF}}}}{2 \cdot \#(MV(m))}$$

Δ_{REF} is the difference between the frames' indices to which the macroblock m and v_i's target macroblock belongs to, $max\Delta_{\text{REF}}$ is the upper bound on the temporal prediction distance specified at encoding time and $MV(m)$ the set of motion vectors belonging to m. Furthermore, $len(v_i)$ is the length of the motion vector v_i, w_f and h_f are the dimensions of the video and ω_{REF} is used to balance the impact of spatial and temporal distance on Ψ_{mv}.

978-1-4244-8788-2/11/$26.00 ©2011 IEEE

III. Unequal Error Protection Rateless Encoding

In contrast to optimal erasure codes which have to fix the code rate in advance, Digital Fountain codes (rateless codes) are able to produce an infinite number of encoded symbols. This feature comes at the cost of a slight decrease in coding efficiency, which implies their classification as being "near-optimal". On the other hand, the encoding and decoding algorithms are computationally cheap compared to traditional schemes.

The encoding algorithm of LT-codes uses two probability distributions: a so-called *Robust Soliton* distribution which is used to obtain the degrees of encoded symbols and a uniform distribution responsible for the selection of addends (input symbols) contributing to the encoded symbol generation. By replacing the latter with a distribution with $P_\Psi(s_i) = \frac{\alpha+(1-\alpha)\Psi(s_i)}{\sum_j^k \alpha+(1-\alpha)\Psi(s_j)}$ which reflects the perceptual relevance of input symbols s_i of the current source block $\{s_1, \ldots, s_k\}$, the probabilities of successfully decoding perceptually more important input symbols can be raised. In this connection, the appropriate choice of $\alpha \in [0; 1]$ is crucial: a too low value might cause significant transmission overhead due to certain input symbols being ignored by the encoder whereas a too high value may almost eliminate the benefits of perceptual relevance estimation. For decoding, we use a modified version of a binary Gaussian elimination algorithm with a matrix of non-fixed size as data structure—increase and decrease of its dimension is caused by incoming encoded symbols and resolved (covered) input or redundant (released) encoded symbols respectively.

IV. Performance Evaluation

We evaluate the performance of the proposed scheme in a small testbed consisting of three Linux boxes serving as sender, receiver and intermediate node respectively. For transmitting data, we use the unreliable transport protocol DCCP which determines the maximum allowed sending rate based on the connection's round-trip time and its packet-loss rate. These parameters are adjusted using Netem [6] at the intermediate node. Once the receiver managed to reconstruct all input symbols, it notifies the sender via a feedback mechanism to avoid unnecessary traffic. In the context of this paper, we assume a non-adaptive sending behaviour i.e. the sender transmits at the maximum allowed rate whenever a non-ACKed source block is pending. Source blocks get either ACKed or are removed from the sending queue once their deadline is reached. New source blocks are fed to the sender according to display timestamps of frames they contain. With regard to the rateless coder, slices encapsulated as NAL units are considered as input symbols and GOPs are handled as source blocks.

First test results prove the positive impact of our scheme and video quality measurements indicate an increase of up to 2.3dB compared to traditional rateless encoding schemes. Especially in cases where only a small fraction of symbols was not decodable, a considerable enhancement in video quality could be observed. Due to space constraints, we omit the summary of all test results and instead discuss one selected instance: we transmitted a H.264 encoded high-resolution 300-frame sequence having a maximum IDR interval of 24 frames and limit the slice size to 1400 Bytes to facilitate their mapping onto network packets. A rather high packet loss ratio of 5% with a burst probability of 20% and a normally distributed RTT (μ=60 ms, σ=6 ms) was used and our scheme was initialized with the following parameter set: ω_{type}=0.39, ω_{dep}=0.36, κ=0.4, ω_{inv}=0.25, ω_{REF}=0.652, α=0.2. The average bit rate of the sequence was 312 kByte/s which occasionally exceeded the connection's maximum transfer rate and thereby contributed to the loss of 7.8% of all NAL units. As depicted in Figure 1,

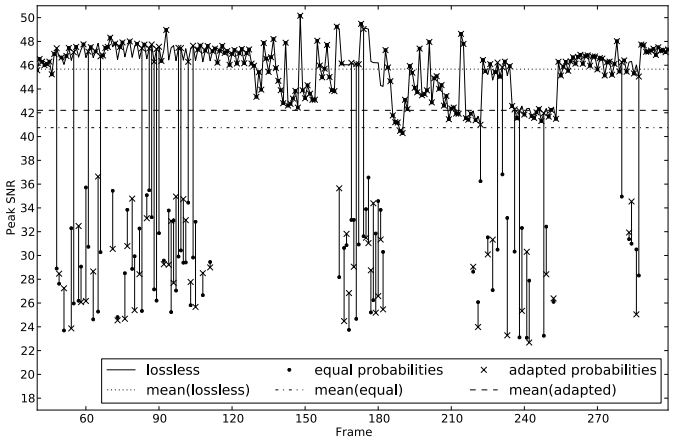

Fig. 1. Quality degradation caused by uncovered input symbols

the mean PSNR could be increased by 1.6dB by adapting the probability distribution as proposed in Section III.

When considering individual loss-affected source blocks, an interesting observation is that although nearly the same number of input symbols could not be decoded in both schemes, a higher number of frames is affected when using an equal distribution for selecting contributing symbols. This phenomenon is directly related to the influence of Ψ_{dep}. With regard to coding efficiency, we could observe a slight increase of the number of necessary symbols of up to 3.5%. Besides combining network-specific feedback provided by the DCCP protocol and content-specific information to optimize the sending schedule and rate, a central aim of further research activities is to mitigate this side-effect e.g. by adapting the degree distribution.

References

[1] M. Luby, "LT Codes," in *Proceedings of the 43rd Symposium on Foundations of Computer Science*. IEEE Computer Society, 2002, p. 271.

[2] M. Schier and M. Welzl, "Selective Packet Discard in Mobile Video Delivery based on Macroblock-Level Distortion Estimation," in *IEEE Infocom MoViD Workshop*, April 2009, pp. 1–6.

[3] N. Rahnavard, B. N. Vellambi, and F. Fekri, "Rateless Codes With Unequal Error Protection Property," *IEEE Transactions on Information Theory*, vol. 53, no. 4, pp. 1521–1532, April 2007.

[4] A. Talari and N. Rahnavard, "Unequal Error Protection Rateless Coding for Efficient MPEG Video Transmission," in *IEEE Military Communications Conference*, October 2009, pp. 1–7.

[5] *H.264 - Advanced Video Coding for Generic Audiovisual Services*, Telecommunication Standardization Sector ITU Std., March 2010.

[6] S. Hemminger, "Network Emulation," in *Linux Conf Au*, April 2005.