MULTI SOURCE STREAMING FOR ROBUST VIDEO TRANSMISSION IN MOBILE AD-HOC NETWORKS

Thomas Schierl[1], Cornelius Hellge[1], Karsten Gänger[1], Thomas Stockhammer[2] and Thomas Wiegand[1]

[1] Fraunhofer Institute for Telecommunications
– Heinrich Hertz Institute,
Einsteinufer 37, 10587 Berlin, Germany
{schierl|gaenger|hellge|wiegand}@hhi.fhg.de

[2] Nomor Research GmbH
Tannenweg 25
83346 Bergen, Germany
stockhammer@nomor.de

ABSTRACT

Video transmission over networks is getting increasingly important, thus upcoming network types like Mobile Ad-Hoc Networks (MANETs) can also become a suitable platform for exchanging/sharing real-time video streams. We present an approach using different sources with linear independent representations of a layered video stream for increasing the robustness in transmission. This approach is based on the Scalable Video Coding (SVC) Extensions of H.264/MPEG-4 AVC with different layers for assigning importance for transmission. Additionally a novel Unequal Packet Loss Protection (UPLP) scheme based on Raptor forward error correction codes is employed. This scheme allows for reception from different sources and benefits from it. While the reception of a single stream guarantees base quality at least, the combined reception enables play-back of video of full quality and/or lower error rates.

***Index Terms*—** Channel coding, Video coding, Multimedia systems, Real time systems, Transport protocols, Network reliability

1. INTRODUCTION

Network infrastuctures based on the Wireless LAN (WLAN) 802.11 a,b,g – specifications are becoming more and more popular and are also suitable for real-time transmission. These network types allow very high data-rates, but usually only in smaller ranges, i.e. as long as the distance between receivers and senders is sufficiently small. Longer ranges and coverage extension can for example be achieved by multi-hop transmission: In contrast to the common setup where many nodes are available in a certain area with a single access point or hot-spot based network structure, the Ad-Hoc mode of WLAN nodes can support the increase of coverage and thus may save costs in infrastructure. These so called Mobile Ad-Hoc Networks (MANETs) [1] make use of all nodes in a mobile network as routers in order to establish a short-time dynamic network infrastructure. An example MANET is shown in Fig. 1 with one client and different established routes (colored) to multiple server nodes.

However, despite the advantages in coverage, the dynamic behavior of MANETs still imposes significant challenges to high quality real-time video transmission: The high path outage probability makes it impossible to reliably transmit a video stream by simply applying known techniques from the wired or cellular transmission environment. Therefore, this work proposes a robust multi-source video streaming protocol for reliable real-time video streaming, which mainly solves the route-loss problem in case of real-time streaming over MANETs. The basic approach to enhance reliability relies on the use of different streaming sources in parallel. Thereby, each source generates independent representations of on and the same media layer. This scheme effectively results in a network multiple description approach. To realize this property the video stream is divided into different layers using Scalable Video Coding (SVC) [2][3]. SVC provides layers with different importance for the video reconstruction and different percentage of the complete stream bit-rate. For effective combination an unequal packet loss protection (UPLP) scheme based on recently proposed Raptor Forward Error Correction (FEC) codes [4] is proposed to protect different layers with different importance. Thereby, the fountain property of Raptor codes enables the generation of a virtually infinite amount of independent Encoding Symbols (ESs) from a fixed number of Source Symbols (SSs). Transmitting these ESs adequately over different paths, preferably from different sources, strongly enhances the reliability of streaming sessions in MANETs. Luckily, the proposed scheme the servers does not require synchronization or coordination among streaming sources, as with high probability by pure randomness independent ESs per source node are generated.

This paper is related to peer-to-peer networks approaches like Avalanche [5] which is developed for file sharing download services and also comprises multi-source approaches. However, our approach is targeting real-time streaming with the addition of being deployed in MANETs. Previous work in this area for wired Internet has for example been presented in [6], but our approach is targeting independently decode-able network streams.

2. MULTI-SOURCE STREAMING COMPONENTS

This section provides a brief overview of the employed components. Their relationship to our proposed multi-source streaming approach is mainly presented in Section 3.

2.1. Real-Time Media Delivery in MANETs

In MANETs each node operates either as a component of the mesh, as duplex mobile node (forwarding), as regular data consuming node, as data generating node, or as a router. Especially the router functionality is what makes MANETs different to wired networks: The issue of continuous routing has to be solved. Most apparently, as in such a distributed system, individual nodes do not have the full knowledge of the complete network's topology, the routing decisions must be made ad hoc and are in general suboptimal when compared to full knowledge wired networks.

One of the key characteristics of MANETs is the excessive time-variant behavior especially in case the participating nodes are mobile resulting in quite unreliable wireless transmission conditions. Another aspect of time variance is the sporadic participation of individual mobile nodes in the network. Therefore, it is quite difficult to set up and maintain the shortest route to a specific destination within the MANET. Route changes as well as route breakdowns occur quite frequently.

2.2. Scalable Video Coding (SVC)

SVC [2][3] is an extension to the H.264/MPEG4-AVC video coding standard. The basic idea of SVC is to extend the hybrid video coding approach in a way that a wide range of spatio-temporal and quality scalability is achieved. Scalability within SVC is a functionality that allows the removal of parts of the bit-stream while achieving a reasonable coding efficiency of the decoded video at reduced temporal, SNR, or spatial resolution. An SVC bit-stream consists of a base layer and one or several nested enhancement layers. The base-layer is a plain H.264/MPEG4-AVC bit-stream ensuring backward-compatibility to existing receivers.

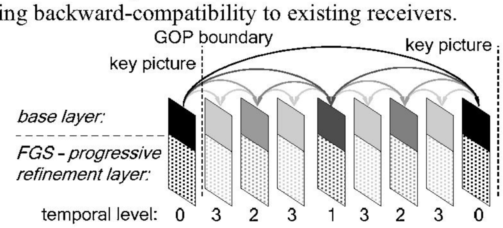

Fig. 1 – Temporal structure of an SVC stream incl. FGS

The temporal scaling functionality of SVC is typically based on a temporal decomposition using hierarchical B pictures as shown in Fig. 1. Each B picture of a higher temporal enhancement level is encoded with a higher QP (cascaded QP assignment), thus the fidelity per picture is decreasing with the decreasing importance in terms of the number of succeeding references by other pictures. In Fig. 2 the structure of a SVC stream is shown, which comprises a group of pictures (GOP) of size eight. GOPs can be independently decoded, if the preceding key picture is available and has random access properties.

Spatial scalability is achieved by different encoder loops with an over-sampled pyramid for each resolution (e.g. QCIF, CIF, and 4CIF), including motion-compensated transform coding with independent prediction structures for each layer. In contrast to the encoder, the decoder can be operated in single loop, i.e., for decoding inter-layer dependencies it does not need to perform motion compensation in lower layers on which it depends.

SNR scalability is based on a Progressive Refinement (PR) approach, where the extension layers contain refinement textural quality information of the base layer in a progressive way. Thus, cutting byte-wise from the end of a PR fragment is possible.

Within this work Temporal, Spatial and SNR scalability based on PR is used for differentiation between video layers. A certain combination of these scalability values is forming an operation point of the layered bit-stream.

2.3. Raptor Forward Error Correction Codes

The Raptor code [4] (Fig. 2) is an error correction code mainly used in environments with packet losses. Furthermore, it is a fountain code, i.e. a virtually infinite amount of encoding symbols can be produced from a vector of source symbols *SV* of the length k. In average, the decoder is capable of reconstructing the source symbols from a number of *ES* that is only slightly higher than the original length of the *SV*. For sufficiently long k, the Raptor code operates very closely to an ideal fountain code which would require only any k encoding symbols for successful reconstruction.

The Raptor code can be viewed as a serial concatenation of a pre-code and a Luby Transform (LT) code. The inner LT code [4] is a realization of a fountain code. From an input vector F an infinite number of encoding symbols ES_i can be produced by XORing symbols in F as indicated in vector $\mathbf{\Psi_i}$, i.e. the vector $\mathbf{\Psi_i}$ consists of all 0's except for 1's at the positions to be XORed. The outer systematic fixed rate code $\mathbf{G_p}$ is the key for improved performance of Raptor codes over LT codes. It is important that the vectors $\mathbf{\Psi_i}$ are generated according to a random distribution whereby the degree distribution and the pre-code need to be appropriately selected for optimized performance.

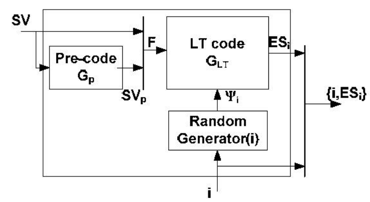

Fig. 2 – Non-Systematic Raptor Encoder

For a maximum likelihood Raptor decoder, a system of linear equations needs to be solved to reconstruct the k source symbols. More precisely, it is sufficient to reconstruct the vector F of intermediate symbols of length $k+s$ as the first k symbols of F correspond to the source symbols. With $\mathbf{1}_s$ being the identity matrix of size s, the received symbols vector $[ES_{i1},\ldots, ES_{ir}]$ of length r, and $\mathbf{G_{LT}}(i_1..i_r)$ the corresponding generator matrix contains the vectors $\mathbf{\Psi_i}$ as the rows. The decoder needs to solve the following equation:

$$\begin{bmatrix} \mathbf{G}_p & \mathbf{1}_s \\ \mathbf{G}_{LT}(i_1..i_k) & \end{bmatrix} \cdot \mathbf{F} = \begin{bmatrix} \mathbf{0}_S^T ES_{i_1}^T .. ES_{i_k}^T \end{bmatrix} \quad (1)$$

Raptor codes are constructed such that (1) can be solved for r equal or only slightly larger than k. Note however, to obtain $\mathbf{G_{LT}}(i_1, \ldots, i_r)$ with each symbol ES_i the symbols index i to generate $\mathbf{\Psi_i}$ needs to be available at the receiver. A systematic version of Raptor codes has been adopted by 3GPP [7] for download as well as for streaming delivery services in MBMS. For the systematic version of this code an additional pre-coding matrix of length k is used on the *SV* to ensure, that the first k of n *ES*s are equal to the *SV*. The Raptor symbol size is in general T bytes. To simplify the simulations, the performance of Raptor codes can be quite well emulated [7]. This model is applied for simulations in this work.

3. MULTI SOURCE STREAMING

This section introduces the proposed multi source streaming approach by appropriately combining the components introduced in section 2. The basic idea is to use multiple sources for guaranteeing reliability by redundancy in sources as well as in network paths. Nevertheless the selected network paths are mainly determined by the underlying MANET routing protocol. The proposed approach of this work is exclusively relies on application layer techniques and does not need any special cross-layer interfaces.

3.1. Media and channel coding

By combining layered video coding with a Raptor code-based

Unequal Packet Loss Protection (UPLP), a distribution of linearly independent representations of the video stream among servers can be achieved. Fig. 3 illustrates the behavior of the proposed multi-source streaming approach using the Raptor code for generating S linear independent network streams from L layers from S sources.

Source / Sender behavior:

The idea is that each source generates linear independent Raptor Encoding Symbols (ESs) for data corresponding to a certain amount of media time t_{SB}. This data, referred to as Source Block (SB), is divided in a certain number of symbols of a of each media layer l, i.e. each source s generates and emits $\hat{n}_l^s$ ESs from k_l source symbols of layer l for a media time interval t_{SB},. That means a source s sends in total a number of

$$\hat{N}_L^s = \sum_{l=1}^{L} \hat{n}_l^s \tag{2}$$

symbols per time interval t_{SB}, as expressed in (2). With a symbol size T_l in bytes for layer l, we obtain the transmission rate $\hat{r}_{tl}^s$ for layer l at source s in terms of bytes per second as

$$\hat{r}_{tl}^s = \frac{\hat{n}_l^s \cdot T_l}{t_{SB}} \tag{3}$$

Assuming limited transmission rate $\hat{R}_t^s$ for source s the transferable symbols per layer l within time interval t_{SB}, is constrained by

$$\hat{R}_t^s \geq \sum_{l=1}^{L} \hat{r}_{tl}^s = \frac{1}{t_{SB}} \sum_{l=1}^{L} \hat{n}_l^s \cdot T_l \tag{4}$$

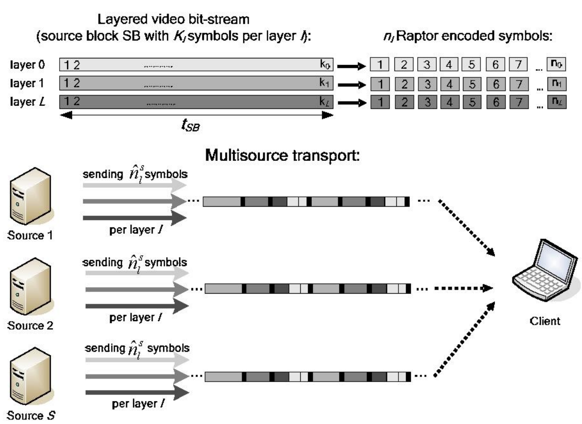

Fig. 3 – Multi-source transport of layered media

Sink / Client behavior:

Assume now that a client receives for a time interval t_{SB} a number of $\tilde{N}_l^S$ linearly independent symbols from S sources for some layer l. If Raptor decoding is successful which is usually the case if the number of received symbols $\tilde{N}_l^S$ is equal to or slightly higher than k_l decoding is successful. Assuming an average, layer independent loss rate λ_s the expected number of received symbol for layer l is computed as

$$\tilde{N}_l^S = \sum_{s=1}^{S} \tilde{n}_l^s = \sum_{s=1}^{S} (1-\lambda_s)\hat{n}_l^s \tag{5}$$

That means a client can affect the number of received symbols per layer, by selecting the number S of sources. All other parameters are basically determined by the transmission scheme.

Reasonable constraints for sender and UPLP process:

Still to be selected are transmission parameters. in this work we apply some reasonable settings rather than looking for detailed optimization. This is subject of ongoing work.

At first, the characteristic of the layered media is defined in terms of byte-rate r per layer. With a constant media rate per SB, the media rate R_L as well the parameter t_{SB}, k_l and T_l are connected as

$$R_L = \sum_{l=1}^{L} r_l = \frac{1}{t_{SB}} \sum_{l=1}^{L} k_l \cdot T_l \tag{6}$$

The symbol arrangement for the UPLP approach per source is according to the example as shown in Fig. 4. The interleaved combination of S source streams is forming the Encoding Block (EB), which contains the ESs representing the media of a time interval t_{SB} of Source Block SB.

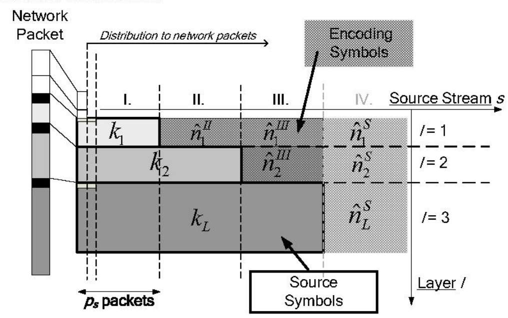

Fig. 4 – Symbol distribution in an encoding block (EB)

The quotient of source and encoding symbols is given by the code-rate $r_{c\,l}$ for a layer l and source s is given as

$$r_{cl} = \frac{k_l}{\hat{n}_l^s} \tag{7}$$

For simplification we assume that the streaming server transmission rate $\hat{R}_{tl}^s$ is fixed for all sources s and with that also the number of transmitted symbols $\hat{n}_l^s$. Proportionally to the code rates $r_{c\,l}$ per layer l, the symbols $\hat{n}_l^s$ are distributed into UPLP packets as shown in Fig. 5.

A reasonable setting for code rates $r_{c\,l}$ is proposed by

$$r_{cl} = \frac{L-l+1}{L} \tag{8}$$

With that constraint it is guaranteed, that with each additional received source stream s+1 the next layer l+1 becomes decodeable at the client with very high probability and the rationales of applying decreasing code rates for more important layer with lower layer indices l is employed. With the constraints mentioned above, the proposed transmission rate (not considering packet loss) per source s is given as

$$\hat{R}_t^s = \sum_{l=1}^{L} \hat{r}_{tl}^s = \sum_{l=1}^{L} r_{cl} \cdot r_l \tag{9}$$

The overall multi source coded stream rate as fraction of R_L (compared to the rate of the original layered media) included by the approach is mainly determined by the rate proportions of the layered media itself. In general the overhead fraction x is shown in (10). Optimizations will follow in later work based on [6].

$$x = \frac{1}{R_L} \sum_{l=1}^{L} (L-l) r_l \tag{10}$$

The source block size k_l and symbol size T_l for each layer l is chosen according to the constraints in [4][7], such that the raptor coding process for each layer l can be assumed as being almost identical to ideal coding as

$$\frac{r_l}{T_l} t_{SB} \geq k_{\min} \quad (11)$$

A key feature of the approach is to guarantee that the different sources are using different random seeds i for generating $\mathbf{\Psi}_i$ for the Raptor encoding process ensuring the generation of independent ESs for each network/source stream. This allows a huge number of sources for generating equivalent but linear independent network streams. Therefore, the sources do not need coordination. A source is selecting randomly a value from a set I of valid values of i for Raptor encoding. I is known to all encoders and decoders and should be of reasonable size guaranteeing independence of ES with high probability.

3.2. Protocol Proposal for Multi-Source Media Delivery

A protocol has been developed for requesting streams from multiple sources and for monitoring quality indicators (metrics) for source node connections. The protocol is implemented in ns-2 and relies exclusively on application layer techniques. The transmission is completely RTP based.

The protocol is entirely receiver-driven. A video client probes all accessible source nodes by sending *inquiry packets* in order to collect information about the network. To keep a consistent status about the network condition and accessible video sources, it is proposed to probe the network continuously during media transmission. All required information for link quality evaluation is transferred and evaluated on the application layer using RTCP packets and header extensions to RTP packets.

By analyzing the distance (in hops) as metric (link quality) information, the best servers (from the protocol point of view) for a multi source streaming session are identified and streams are requested from those servers which appear to have the shortest metric. The assumption is: A better metric (shorter path) enables a smaller connection of failure probability and packet loss probability due to the addition of loss probability of each link of the used path in a mobile ad-hoc network. We refer to further work of the authors for more detailed information [8].

4. SELECTED SIMULATION RESULTS

All network simulations have been carried out with ns-2. The video streams have been encoded with version JSVM5 of the SVC reference software. A repeated Foreman sequence with 8640 frames (285 sec), with GOP size 16, random access property and three operation points is used:

	Res.	F-rate/fps	Bitrate/kbps	PSNR/dB (upscaled)
1	QCIF	15	42.96	32.62 (27.82)
2	CIF	15	155.69	31.75
3	CIF	30	438.48	37.16

Tab.1: Operation points of SVC media stream.

The stream structure allows for generating UPLP streams with three protection classes. The resulting overall multi-source coded bit-rate using UPLP for three source streams includes about 47% redundancy compared to the original media bit-stream, that allows for reception from minimal 3 different sources. DYMO [9] is used as underlying MANET routing protocol in combination with IEEE 802.11b. In an area of 800m x 800m, 19 nodes are moving in the presence of 1 client and up to 4 available serves. The physical distance of the nodes allows a maximum wireless network rate of 1MBit/sec per hop air-interface.

The results are evaluated in a fixed scenario with a certain link-loss characteristic per source and with an average source availability (in percentage of simulation time, where server is not in link-loss state) of approximately 50% per available server.

The overall results are shown in terms of received video quality (Peak signal-to-noise ratio (PSNR) and play-able frame-rate) for using different available sources in the evaluation scenario in Tab.2.

mode	*Orig.*	*Single*	*Multi*	*Multi*	*Multi*
avail. Sources	-	1	2	3	4
avg. PSNR(dB)	37.16	23.46	31.24	36.16	36.46
avg. F-rate(fps)	30	12.62	20.93	25.65	27.96

Tab.2: Overall quality results for using different number of sources

The results show that the proposed multi source approach outperforms the single server approach (which is compare-able with a state-of-the-art streaming technologies) for both quality indicators.

5. CONCLUSION

We presented an extended Unequal Packet Loss Protection (UPLP) scheme based on Raptor FEC using different sources for reliable media streaming in MANETs. We showed the benefits of using linear independent FEC streams with unequal loss protection for Multi Source Streaming in scenarios with high route loss probability as present in MANETs. The approach has been tested with a new protocol for media tracking and delivery in MANETs, which exclusively relies on Application Layer techniques. For more details of the approach, additional and more comprehensive results as well as other extensions we refer to [8]:

6. REFERENCES

[1] IETF Mobile Ad-Hoc Network (MANET) working group, http://www.ietf.org/html.charters/manet-charter.html

[2] H. Schwarz, D. Marpe, T. Schierl, and T. Wiegand, "Combined Scalability Support for the scalable Extensions of H.264/AVC," IEEE ICME, July 2005

[3] T. Wiegand, G. Sullivan, J. Reichel, H. Schwarz, and M. Wien (eds.), "Joint Draft 5: Scalable Video Coding (in integrated form with ITU-T Rec. H.264 | ISO/IEC 14996-10)," Joint Video Team (JVT), Doc. JVT-R201, January. 2006

[4] M. Luby, T. Stockhammer, M. Watson, T. Gasiba, and W. Xu "Raptor Codes for Reliable Download Delivery in Wireless Broadcast Systems," IEEE CCNC, January 2006

[5] C. Gkantsidis and P. R. Rodriguez "Network Coding for Large Scale Content Distribution," IEEE Infocom, March 2005

[6] T. Nguyen and A. Zakhor, "Distributed Video Steaming using Forward Error Correction," PVW, April 2002

[7] 3GPP TS 26.346 V6.4.0, "Technical Specification Group Services and System Aspects; Multimedia Broadcast / Multicast Service (MBMS)", March 2006

[8] T. Schierl, T. Stockhammer, T. Wiegand, „SVC-based Multi Source Streaming for Robust Video Transmission in Mobile Ad-Hoc Networks", Special Issue, Multimedia in Wireless/Mobile Ad-hoc Networks, IEEE Wireless Communications Magazine, accepted Paper, 2006

[9] I. Chakeres, E. Belding-Royer, and C. Perkins, "Dynamic MANET On-demand (DYMO) Routing," draft version 04, IETF, March 2006