Rate-distortion Optimized Network Coding for Cooperative Video Stream Repair in Wireless Peer-to-Peer Networks

Xin Liu
University of California, Davis
xyzliu@ucdavis.edu

Gene Cheung
HP Labs Japan
gene-cs.cheung@hp.com

Chen-Nee Chuah
University of California, Davis
chuah@ucdavis.edu

Abstract

By providing coding ability at intermediate nodes, network coding has been shown to improve network throughput in broadcast/multicast wireless networks. In this paper, we show that by imposing coding structure, network coding can be further optimized specifically for video streaming in a rate-distortion manner, in a scenario where wireless adhoc peers cooperatively relay packets to each other to repair packet losses during MBMS broadcast. Experimental results show that our proposed scheme can improve video quality noticeably, by up to $19.71dB$ over un-repaired video stream and by up to $7.90dB$ over video stream using traditional unstructured network coding.

1 Introduction

Due to wireless cellular networks' limited bandwidths and unreliable transmission channels, delivery of high quality video over these networks has long been known to be a difficult problem [1]. The advent of Multimedia Broadcast Multicast Service (MBMS) [2], in 3GPP specification version 6 or later, means media content can now be delivered to multiple users simultaneously and efficiently via a shared channel. However, it also means previously developed feedback-based packet loss recovery schemes [1] for point-to-point streaming are no longer applicable due to the well-known NAK implosion problem, making video delivery over MBMS an even more difficult task.

To address the problem, we have previously proposed a *Cooperative Peer-to-Peer Repair* (CPR) framework [3] for a community of wireless peers with both cellular and 802.11 network interfaces. The idea is simple: having each correctly received a different subset of packets from MBMS broadcast (due to different channel conditions experienced), a local cluster of nodes can then locally broadcast their packets via 802.11 to cooperatively recover lost packets. Using our developed heuristics, we showed in [3] that significant packet recovery can be achieved under reasonable network settings. Moreover, if we permit each peer to perform *Network Coding* [4]—linearly combining payloads of received packets in $GF(2^O)$, where 2^O is the field size—before forwarding packets, we showed in [5] that even further performance gain in packet recovery can be achieved.

Compared to its cellular counterpart, a 802.11 network requires much more power to establish and maintain connections [6, 7]. Therefore powering both interfaces continuously for the entire duration of a long video stream is neither energy-efficient nor practical for a lightweight, battery-powered mobile device. For the purpose of CPR packet recovery then, it is more sensible to instead activate the 802.11 interface for only duration τ in every period T, where τ and T together determine the fraction of 802.11 bandwidth available for peer-to-peer packet transmissions. In this energy-limited scenario, the more challenging research problem is the following: given a fraction of 802.11 bandwidth available for a limited number of peer-to-peer packet transmissions, how to perform cooperative packet repair at each peer so that the expected video distortion at the average peer is minimized?

In this paper, we present a novel rate-distortion optimized, network-coding based, cooperative video stream repair scheme for the energy-limited scenario. Unlike typical network coding schemes, we structure network coding so that packets of important frames can be recovered with appropriately higher probabilities than less important ones. Experiments showed that our structured network coding scheme improves video quality by up to $19.71dB$ over unrepaired video stream, and by up to $7.90dB$ over video stream using unstructured network coding scheme.

The outline of the paper is as follows. In Section 2, we overview related works. In Section 3, we discuss our chosen source and network models. We differentiate unstructured and structured network coding, the latter of which is used in our optimization framework shown in Section 4. Based on these discussions, we present our optimization framework in Section 5. We explain our experimentation and discuss the results in Section 6. We conclude in Section 7.

978-1-4244-2100-8/08/$25.00 ©2008 IEEE

2 Related Work

Due to the aforementioned NAK implosion problem, many video streaming strategies over MBMS [8] have forgone feedback-based error recovery schemes like [1] and opted instead for Forward Error Correction (FEC)-based schemes like Raptor Codes. While FEC can certainly help some MBMS receivers recover some packets, receivers experiencing transient channel failures due to fading, shadowing, and interference can still suffer great losses. Nevertheless, content source should perform some optimization to lessen the loss impact. In this work, we assume content source will first perform *reference frame selection* [9] during H.264 [10] video encoding so that inter-frame dependencies are minimized subject to an encoding rate constraint.

Network coding [4] has been an active research topic, and recent works [11, 12, 13] have attempted to jointly optimize video streaming and network coding. [12] discussed a rate-distortion optimized network-coding scheme on a packet-by-packet basis for a wireless router, assuming perfect state knowledge of its neighbors. Though the context of our CPR problem is different, our formulation can be viewed as a generalization in that our optimization is on the entire Group of Pictures (GOP), while [12] is performed greedily per packet. [11] discussed a hierarchical network coding scheme where a layered structure is applied to a scalable, layer-coded video stream. Our formulation is more general in that our source dependency graph is a directed acyclic graph discussed in the next section while the model in [11] is essentially the more restricted dependency chain. [13] discusses the application of Markov Decision Process [14] to network coding, in which network coding optimization and scheduling are directed centrally at the access point or base station. They require complete state information assuming reliable ACK/NAK schemes which has not been shown to be scalable to large number of peers. In our work, we consider fully distributed peer-to-peer repair without assuming any knowledge of state information of peers, and instead optimize using a pre-determined network coding structure.

3 Models and Reference Frame Selection

3.1 Source Model

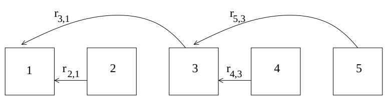


Figure 1. DAG Source Model for H.264 Video with Reference Frame Selection

We assume the content source first performs *reference frame selection* during encoding of H.264 video [9] such that the inter-dependencies of frames in a GOP is minimized. In brief, the optimization works as follows. We first assume that each GOP is composed of a starting I-frame followed by $M-1$ P-frames. Each P-frame can choose among a set of previous frames for motion compensation (MC), where each choice results in a different encoding rate and different dependency structure. If we now assume that a frame is correctly decoded only if it is correctly received and the frame it referenced is correctly decoded, then this choice also leads to a different correctly decoded probability. Using P-frames' selections of reference frames, [9] seeked to maximize the expected number of correctly decoded frames given an encoding rate constraint.

Note that though H.264 [10] specification is more general and permits each coding block in a P-frame to individually choose a matching block in one of a number of previous frames for MC, we restrict all blocks in a given P-frame to point to a single previous frame. [9] showed that the streaming benefit outweighted the cost in coding restriction.

After the content source performed reference frame selection, we can now model M frames in a GOP, $\mathcal{F} = \{F_1, \ldots, F_M\}$, as nodes in a *directed acyclic graph* (DAG) as shown in Figure 1, similarly done in [14]. Each frame F_i has an associated d_i, the resulting distortion reduction if F_i is correctly decoded. Each frame F_i points to the frame in the same GOP using which F_i performs MC. A frame F_i is correctly decoded iff F_i is correctly received by its decoding deadline, *and* all frames F_j's preceding F_i, $j \prec i$, are correctly decoded. Frame F_i referencing frame F_j results in encoding rate $r_{i,j}$. We assume each frame F_j is packetized into multiple RTP packets according to the frame size and Maximum Transport Unit (MTU) of the delivery network.

3.2 Network Model

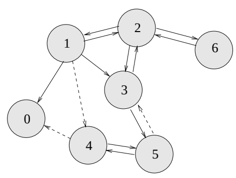


Figure 2. Directed Graph Network Model: transmission and interference links are solid and dotted lines, respectively.

As done in [5], we assume that N peers in a wireless peer-to-peer network are modeled by nodes $1, \ldots, N$ in node set $\mathcal{N}$ in a directed graph $\mathcal{G} = (\mathcal{N}; \mathcal{L}_T, \mathcal{L}_I)$, and connectivities and interferences among nodes are modeled by links in link sets $\mathcal{L}_T$ and $\mathcal{L}_I$, respectively. See Figure 2 for an example. A peer n_2 correctly receives a packet from transmitting peer n_1 iff: i) there exists a transmission link from n_1 to n_2, i.e., $(n_1, n_2) \in \mathcal{L}_T$; and ii) no other nodes whose transmission or interference ranges include n_2, i.e., $\forall n_i | (n_i, n_2) \in \mathcal{L}_T \cup \mathcal{L}_I$, is transmitting at the same time as n_1. Notice that by this definition of successful transmission, we implicitly imply that the *broadcast mode* of 802.11 is used, where the transmission of a node can potentially be heard by all its neighbors.

Although the raw transmission rate of 802.11 is large, the peers need to contend for the shared medium for transmission in some distributed manner so that the occurrences of collision (simultaneously transmission of two nodes n_1, n_2 to a third node n_3 where $(n_1, n_3), (n_2, n_3) \in \mathcal{L}_T$) and interference (simultaneously transmission of n_1, n_2 where $(n_1, n_3) \in \mathcal{L}_T$ and $(n_2, n_3) \in \mathcal{L}_I$) are reduced. Note that while transmission links $\mathcal{L}_T$ are discovered through local message exchanges, interference links $\mathcal{L}_I$ are unknown to peers. To avoid collisions and interferences, we assume each peer n performs a collision avoidance procedure at the MAC layer: a clocking mechanism that backs off a varying random amount of time when it senses the carrier is busy.

4 Network Coding for CPR

In this section, we describe how network coding can be used in the context of CPR. In particular, beyond the well known *Unstrcutured Network Coding*, we present *Structured Network Coding*, which can be optimized for video streaming in a rate-distortion manner.

4.1 Unstructured Network Coding

Suppose there are M original (native) frames $\mathcal{F} = \{F_1, \ldots, F_M\}$ to be disseminated among N peers in a CPR setting. Each frame F_k is divided into multiple packets $\mathcal{P}_k = \{p_k^1, p_k^2, \ldots, p_k^{B_k}\}$, where B_k is the number of packets frame F_k is divided into. We use $\mathcal{P}^*$ to denote the set of all the packets in a GoP, i.e., $\mathcal{P}^* = \{\mathcal{P}_1, \ldots, \mathcal{P}_M\}$. Therefore there are $P = |\mathcal{P}^*| = \sum_{i=1}^{M} B_i$ packets to be disseminated. Using network coding (NC), each peer n can generate and transmit a NC packet q using a linear combination of its set of received MBMS native packets $\mathcal{G}_n$ and its set of received NC packets $\mathcal{Q}_n$ as follows:

$$q = \sum_{p_k^i \in \mathcal{G}_n} a_k^i p_k^i + \sum_{q_m \in \mathcal{Q}_n} b_m q_m \quad (1)$$

$$= \sum_{p_k^i \in \mathcal{P}^*} c_k^i p_k^i, \quad (2)$$

where a_k^i's and b_m's are random numbers in $GF(2^O)$. a_k^i is the random coefficient for each of the original packet and b_m is the random coefficient for the received encoded NC packet. Because each received NC packet q_m is itself a linear combination of native and NC packets, we can rewrite q as a linear combination of native packets with *native coefficients* c_k^i's as shown in (2). For unstructured network coding (UNC), a_k^i's and b_m's are always non-zero, and a peer can reconstruct all P native packets when P "innovative" native or NC packets are received and therefore all the frames can be recovered. By innovative, we mean that native coefficient vector $\mathbf{v} = [c_1^1, \ldots, c_1^{B_1}, \ldots, c_M^1, \ldots, c_M^{B_M}]$ of a newly received NC packet cannot be a linear combination of native coefficient vectors from the set of previously received innovative native or NC packets. In other words, new native coefficient vector $\mathbf{v}$ must be orthogonal to old native vectors of previous innovative packets.

The downside of UNC is that if a peer n receives fewer than P innovative native or NC packets, then the peer cannot recover *any* of the native packets from the received NC packets. If the probability of receiving at least P innovative native or NC packets for many peers is low, then this is obviously not a desired result.

4.2 Structured Network Coding

To address the aforementioned issue, we propose to use structured network coding (SNC). By imposing structure in the coefficient vector, we seek to decode at a peer even when fewer than P innovative native or NC packets are received. We accomplish that by forcing some chosen coefficients a_k^i and b_m's to be zeroes during NC packet generation, so that when a peer receives m innovative packets, $m < P$, it can decode m packets (m linear equations for m unknowns). Thus some of the frames can be recovered.

More precisely, given the DAG source model described in Section 3.1, we first define a series of X *frame groups*, $\Theta_1, \ldots, \Theta_X$, where $\Theta_1 \subset \ldots \subset \Theta_X = \mathcal{F}$, and $\Theta_x \subseteq \mathcal{F}$, $1 \leq x \leq X$. Corresponding to each frame group Θ_x is a *NC packet type* x, which is identified in the packet header—$\Phi(q)$ reveals the packet type of NC coded packet q. A peer n then can encode a NC packet $q_n(x)$ of type x, given peer's set of received or decoded native packets $\mathcal{G}_n$ and set of received NC packets $\mathcal{Q}_n$, as:

$$q_n(x) = \sum_{p_k^i \in \mathcal{G}_n} U(F_k \in \Theta_x)\, a_k^i\, p_k^i + \sum_{q_m \in \mathcal{Q}_n} U(\Phi(q_m) \leq x)\, b_m\, q_m, \quad (3)$$

where $U(\mathsf{c})$ evaluates to 1 if clause c is true, and 0 otherwise. In words, peer n constructs NC packet of type x by linearly combining received or decoded packets of frames in Θ_x and received NC packets of type x or smaller.

A peer n can recover all m packets in frame group Θ_x once it has received m innovative packets of types $\leq x$. We call this recovery process *NC-decoding*. In the following section, we show how the frame groups are selected using our optimization framework.

5 Optimization Framework

5.1 Optimization Formulation

We assume a video source using MBMS delivers each GOP of M frames in time duration T, called an *epoch*. Repairs of the current GOP take place during the next epoch; 802.11 interface of each peer is activated from *sleep mode* to *idle mode* [7] for the first τ seconds of the next epoch T, during which peers can transmit and receive peer-to-peer repair packets of GOP of the previous epoch. The initial playback buffer delay for each peer is therefore two epochs.

Because of the transient join/leave nature of peers in an ad-hoc network, the exact number and connectivities of peers at any given time is difficult to track. Instead, we assume that the video source performs the optimization of the NC structure for the *average peer* n, assuming that on average a peer is expected to have received R_n packets from peers. Using the DAG source model from Section 3.1, the expected distortion at an average peer n can be written as:

$$\Delta_n = D - \sum_{i=1}^{M} d_i \prod_{j \preceq i} \alpha_n(j), \tag{4}$$

where D is the initial distortion of the GOP if no frames are received, and $\alpha_n(j)$ is the recovery success probability of frame F_j at peer n. $\alpha_n(j)$ itself can be written as:

$$\alpha_n(j) = (1-l)^{B_j} + \left(1 - (1-l)^{B_j}\right) S_n(j), \tag{5}$$

where l is the MBMS packet loss rate, and $S_n(j)$ is the probability of frame F_j being recovered at peer n through CPR given F_j was not initially successfully delivered via MBMS. Note we assume that all the packets within F_j must be received in order to decode F_j.

Suppose we are given NC groups $\Theta_1, \ldots, \Theta_X$ with $F_j \notin \Theta_{x-1}$ but $F_j \in \Theta_x$. Then frame F_j can be recovered if $\sum_{F_i \in \Theta_x} B_i$ innovative NC packets of type $\leq x$ are received, or if $\sum_{F_i \in \Theta_{x+1}} B_i$ innovative NC packets of type $\leq x+1$ are received, etc. If a node n sends a NC packet type x with probability $\beta_n(x)$, we can approximate $S_n(j)$ as:

$$S_n(j) = Q(n,x) + \sum_{y=x+1}^{X} Q(n,y) \prod_{z=x+1}^{y} \left(1 - Q(n,z-1)\right), \tag{6}$$

where $Q(n,x)$ is the probability that node n can NC-decode NC packet type x by receiving $\sum_{F_i \in \Theta_x} B_i$ innovative NC packets of types $\leq x$. We approximate $Q(n,x)$ as:

$$Q(n,x) \approx \sum_{k=\left\lceil l \sum_{F_i \in \Theta_x} B_i \right\rceil}^{R_n} \binom{R_n}{k} \left(\sum_{i=1}^{x} \beta_n(i)\right)^{k} \left(\sum_{i=x+1}^{X} \beta_n(i)\right)^{R_n - k}, \tag{7}$$

where $l \sum_{F_i \in \Theta_x} B_i$ is the expected number of lost packets due to MBMS broadcast and to be repaired using CPR. Assuming CPR has perfect collision avoidance, R_n, the average number of packets a peer can receive in an epoch time, can be approximated as:

$$R_n = \frac{\gamma\tau}{L/C_{\max}} \left(\frac{E_n^T}{E_n^T + 1}\right)^{E_n^I}, \tag{8}$$

where γ is the fraction of bandwidth used for packet transmission after collision avoidance, which is estimated via experimentation. L is the average size of a CPR packet. $C_{\max}$ is maximum rate of IEEE 802.11 used for CPR. Therefore $\frac{\gamma\tau}{L/C_{\max}}$ is the maximum number of packets node n can receive during an epoch time without considering interference. $E_n^T = |S : \{\forall n_i | (n_i, n) \in \mathcal{L}_T\}|$ and $E_n^I = |S : \{\forall n_i | (n_i, n) \in \mathcal{L}_I\}|$ are the expected numbers of neighboring and interference nodes of node n, respectively. Both of them are estimated via actual experimentation. We assume each interfering node has the same fraction of time for transmission as its neighboring nodes, thus $E_n^T/(E_n^T+1)$ is the probability that an interfering node doesn't transmit at a given time and $\left(E_n^T/(E_n^T+1)\right)^{E_n^I}$ is the non-interference transmission probability of node n.

With our formulation shown in equations (4)—(8), the SNC optimization process is therefore to find the number of frame groups X, composition of frame groups Θ_x's, and the packet transmission probabilities of NC types $\beta_n(x)$'s corresponding to frame groups so that the average distortion of the GOP is minimized:

$$\min_{X, \{\Theta_x\}, \{\beta_n(x)\}} \Delta_n \tag{9}$$

Next we discuss how the optimization is performed.

5.2 Local Search Solution

We can estimate the size of the search space as follows. Suppose there is only one NC type. Then the number of unique assignments of one NC type to M frames, $K_1(M)$, is trivially 1. Now suppose there are two distinct NC types. Then the number of unique assignments of two NC types to M frames, $K_2(M)$, such that at least two frames have different types, is 2^M combinations minus two single-type assignments where only type 1 or only type 2 is assigned to all frames, i.e., $2^M - 2$. Now suppose there are three distinct NC types. $K_3(M)$ is 3^M minus combinations that assign only two distinct types, minus combinations that assign only one distinct type: $3^M - \binom{3}{2}\left(2^M - 2\right) - \binom{3}{1}(1)$. Since the maximum number of NC types for M frames is also M, the number of unique NC type assignments to M frames, $K(M)$, is:

$$K(M) = \sum_{i=1}^{M} K_i(M) \tag{10}$$

$$K_i(M) = i^M - \sum_{j=1}^{i-1} \left(\begin{array}{c} i \\ j \end{array} \right) K_j(M)$$

For each NC structure, we need to search the transmission probability $\beta_n(x)$ from $(0, 1)$ for each NC type x. Clearly for reasonable values of M—$K(5) = 541$, exhaustive search is not a feasible approach. As such, we present a local search method as follows.

We first notice that the search space can be reduced by considering the DAG structure described in Section 3.1. A frame F_j that precedes frame F_i must surely be as important as frame F_i, since without it F_i cannot be correctly decoded. When we assign frames to NC types then, we will assign preceding frames with a smaller or equal NC type than succeeding frames given the DAG structure.

We perform the local search as follows. We first assign M NC types to the M frames in *topological order* according to the DAG structure, so that a frame F_j preceding F_i will have a NC type smaller than F_i. For this NC structure, we exhaustively search the best $\beta_n(x)$ resulting in the smallest distortion using (9). We then find the best "merging" of parent and child frames—assigning the same NC type to the merged group— according to the DAG, and search for the best $\beta_n(x)$ for each of the group so that the objective is most reduced. We continue until no such beneficial merging operation can be found.

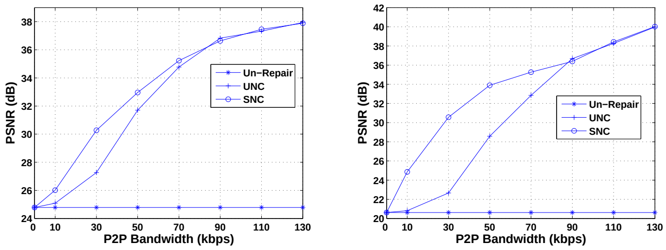
6 Experimentation

6.1 Experimental Setup

We present the benefit of the SNC scheme over the UNC scheme under various CPR transmission rates, i.e., various τ/T ratios. The MBMS source transmits at rate $220kbps$ and the packet loss rate is constant at 0.1. Two test video sequences are used for simulations: 300-frame MPEG class A `news` and class B `foreman` sequences, which are captured at 30fps and sub-sampled in time by 2. The GOP size is 15 frames: one I-frame followed by 14 P-frames. Quantization parameters used for I-frames and P-frames are 30 and 25, respectively. The H.264 codec used is JM 12.4, downloadable from [15]. We perform reference frame selection in [9] with encoding target rate equals $220kbps$, resulting in a DAG describing the inter-frame dependencies as discussed in Section 3.1. The MTU is set to be 1000bytes.

The SNC CPR scheme was performed as follows. With the video source models and the cellular and CPR network parameters, the MBMS source packetizes the frames into multiple RTP packets and searches for the optimal NC structure using the optimization framework shown in Section 5. The MBMS source then adds in the header of each of packets the NC structure. After receiving the packets which are attenuated by the cellular broadcast channel, the peers in the network initiate the packet repairing process using their 802.11 wireless interfaces. Peers exchange their packets according to the pre-determined packet types and transmission probability. The repairing process ends when τ seconds have elapsed.

6.2 Experimental Results



a) `news` b) `foreman`

Figure 3. PSNR for the `news` and `foreman` under various CPR transmission rate.

With our NC frame group optimization using the local search method, it turned out that when the CPR bandwidth was low, the NC group optimization returned more NC types than when the bandwidth was high. The range of the number of NC types was from 2 to 9. We noted that the lower the bandwidth was, the smaller the sizes of the first few NC groups. This is reasonable because when bandwidth was low, peers needed desperately to decode at least the first few frames. Dividing the packets into more groups increased the chance that the received packets could be decoded and therefore peers could at least decrease some of the distortion with the limited number of receiving packets.

Fig. 3a and Fig. 3b show the CPR bandwidth vs PSNR plot for the `news` and `foreman` sequences. The video qualities resulting from the UNC and the SNC schemes were compared, as well as the un-repaired video quality. The CPR bandwidth varied from 0kbps up to 130kbps.

It is clear that both of the UNC and the SNC schemes showed large improvement over the un-repaired video stream. Both of the schemes provided 13.51dB PSNR improvement for the `news` sequence and 19.71dB PSNR improvement for the `foreman` sequence when the bandwidth was larger than 130kbps. For the UNC scheme, the peers needed to possess $\sum_{j=1}^{15} B_j$ innovative native or NC packets before any repairing could be performed. However, for the SNC scheme, nodes could repair part of the frames as long as the received packets could help decode some NC types of frames. This would be much less than the total number of packets. Therefore, when bandwidth was low, i.e., less than 90kbps, the performance of the SNC scheme was much better than the UNC scheme. For example at the transmission rate of 30kbps, the SNC scheme achieved $3.01dB$ gain over the UNC scheme for the `news` sequence and around $7.90dB$ gain for the `foreman` sequence. When the bandwidth was higher, i.e., larger than 90kbps, the number of received packets increased so that the UNC scheme recovered more packets and the performance of the two schemes became similar. Eventually both of the two schemes converged to the same best performance point when the bandwidth was larger than 130kbps, where both of them recovered all of the lost packets.

Comparing Fig. 3a and Fig. 3b, we see similar PSNR trends. However, it is also worthwhile to note that when bandwidth is low, the gap between the SNC and UNC schemes was larger in `foreman` than in `news`. This is due to the fact that `foreman` has more inherent motion and requires more encoding bits for the same given quantization parameters. As a result, the DAG dependency graph was long rather than wide. It means if a particular packet close to the root node is lost, it affects many descendent frames and results in large distortion. Therefore SNC is more important in the `foreman` sequence than the `news` sequence.

7 Conclusions

In this paper, we propose a rate-distortion optimized structured network coding scheme for cooperative video stream repair of MBMS packet losses for 802.11 peer-to-peer networks. We focus on the case when the 802.11 network interfaces are only activated for a short amount of time periodically, and hence the repair bandwidth is low and a limited number of repair packets are transmitted. Specifically, reference frame selection is performed at the content source to minimize inter-frame dependencies in a Group of Pictures in H.264. Packets of video frames are then mapped into a series of frame groups, such that when structured network coding types are designed accordingly and repair packets of these types are sent at each peer with different weights, recovery of frames in more important frame groups are more likely than less important ones. In so doing, we show that our proposed scheme provides as large as $7.90dB$ video quality improvement over the UNC scheme when the CPR bandwidth is low, and up to $19.71dB$ improvement over the un-repaired video stream.

Although the discussion of the paper focused on the network scenario of 802.11 peer-to-peer repair of MBMS broadcast video, a carefully structured NC scheme is also useful for other combinations of peer-to-peer and broadcast technologies. For example, combination of bluetooth based peer-to-peer repair of MBMS broadcast video, or 802.11 peer-to-peer repair of DVB-H broadcast video.

References

[1] G. Cheung, W.-T. Tan, and T. Yoshimura, "Double feedback streaming agent for real-time delivery of media over 3G wireless networks," in *IEEE Transactions on Multimedia*, April 2004, vol. 6, no.2, pp. 304–314.

[2] *Technical Specification Group Services and System Aspects; Multimedia Broadcast/Multicast Service (MBMS) user services; stage 1 (Release 6) (3GPP TS.26.246 version 6.3.0*, March 2006.

[3] S. Raza, D. Li, C.-N. Chuah, and G. Cheung, "Cooperative peer-to-peer repair for wireless multimedia broadcast," in *IEEE International Conference on Multimedia and Expo (ICME)*, Beijing, China, July 2007, pp. 1075–1078.

[4] L. Li, R. Ramjee, M. Buddhikot, and S. Miller, "Network coding-based broadcast in mobile ad-hoc networks," in *IEEE INFOCOM 2007*, Anchorage, AL, May 2007, pp. 1739–1747.

[5] X. Liu, S. Raza, C.-N. Chuah, and G. Cheung, "Network coding based cooperative peer-to-peer repair in wireless ad-hoc networks," in *IEEE International Conference on Communications (ICC)*, Beijing, China, May 2008.

[6] A. Rahmati and L. Zhong, "Context-for-wireless: Context-sensitive energy-efficient wireless data transfer," in *ACM MobiSys*, San Juan, Puerto Rico, June 2007, pp. 165–178.

[7] S. Chandra and A. Vahdat, "Application-specific network management for energy-aware streaming of popular multimedia formats," in *USENIX Annual Conference*, Monterey, CA, June 2002, pp. 329–342.

[8] J. Zfzal, T. Stockhammer, T. Gasiba, and W. Xu, "Video streaming over MBMS: A system design approach," in *Journal of Multimedia*, August 2006, vol. 1, no.5, pp. 25–35.

[9] G. Cheung, W.-T. Tan, and C. Chan, "Reference frame optimization for multiple-path video streaming with complexity scaling," in *IEEE Transactions on Circuits and Systems for Video Technology*, June 2007, vol. 17, no.6, pp. 649–662.

[10] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," in *IEEE Transactions on Circuits and Systems for Video Technology*, July 2003, vol. 13, no.7, pp. 560–576.

[11] K. Nguyen, T. Nguyen, and S.-C. Cheung, "Peer-to-peer streaming with hierarchical network coding," in *IEEE International Conference on Multimedia and Expo (ICME)*, Beijing, China, July 2007, pp. 396–399.

[12] H. Seferoglu and A. Markopoulou, "Opportunistic network coding for video streaming over wireless," in *IEEE 16th International Packet Video Workshop*, Lausanne, Switzerland, November 2007, pp. 191–200.

[13] D. Nguyen, T. Nguyen, and X. Yang, "Multimedia wireless transmission with network coding," in *IEEE 16th International Packet Video Workshop*, Lausanne, Switzerland, November 2007, pp. 326–335.

[14] P. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," in *IEEE Transactions on Multimedia*, April 2006, vol. 8, no.2, pp. 390–404.

[15] "The TML project web-page and archive," http://kbc.cs.tu-berlin.de/ stewe/vceg/.