DISTRIBUTED MARKOV DECISION PROCESS IN COOPERATIVE PEER-TO-PEER REPAIR FOR WWAN VIDEO BROADCAST

Zhi Liu[o#], *Gene Cheung*[#], *Yusheng Ji*[#]

[o] The Graduate University for Advanced Studies, [#] National Institute of Informatics

ABSTRACT

Error resilient video broadcast over Wireless Wide Area Networks (WWAN) remains difficult due to unavoidable packet losses (a result of the underlying unreliable and time-varying transmission medium) and unavailability of per-packet, per-user retransmissions (stemming from the well-known NAK implosion problem). Previous cooperative solutions for multi-homed devices listening to the same video broadcast call for local recovery via packet sharing: assuming peers are physically located more than one transmission wavelength apart, channels to the streaming source are statistically independent, and peers can exchange different subsets of received packets with neighbors via a secondary network like ad hoc Wireless Local Area Network (WLAN) to alleviate individual WWAN packet losses. While it is known that using structured network coding (SNC) to encode received packets before peer exchange can further improve packet repair performance, the decisions of who should send repair packets encoded in what SNC types at available transmission opportunities were not optimized in any formal way. In this paper, we propose a distributed decision making strategy based on Markov decision process (MDP), so that each peer can make locally optimal transmission decisions based on observations eavesdropped on the WLAN channel. Our proposed MDP is both computationally scalable and peer-adaptive, so that state transition probabilities in MDP can be appropriately estimated based on observed aggregate behavior of other peers. Experiments show that decisions made using our proposed MDP outperformed decisions made by a random scheme by at least 4dB in PSNR in received video quality.

Index Terms— Wireless video streaming, cooperative peer-to-peer repair, Markov decision process

1. INTRODUCTION

Video broadcast over Wireless Wide Area Networks (WWAN) remains a challenge, given stringent playback deadlines of real-time video and unavoidable packet losses due to shadowing, channel fading and inter-symbol interference in the wireless medium. Moreover, a streaming server cannot tailor retransmission for every lost packet experienced by each client due to the well-known NAK implosion problem [1]. To overcome channel losses, conventional WWAN video broadcast schemes [2] employ sufficient amount of Forward Error Correction (FEC) packets for channel statistics of a targeted nth percentile user (e.g., 50th percentile or the average user in [2]) to protect source packets. Because of the undesirable "cliff" effect of FEC, however, users with channels worse than the targeted nth percentile user's will experience catastrophic packet losses, resulting in unacceptably poor video quality[1].

[1]Targeting channel statistics of the 0th percentile (worst) user in a large receiver group would mean expending a huge portion of bits out of a fixed WWAN transmission budget for channel coding, leaving few bits for source coding to reduce quantization noise, resulting in poor video quality.

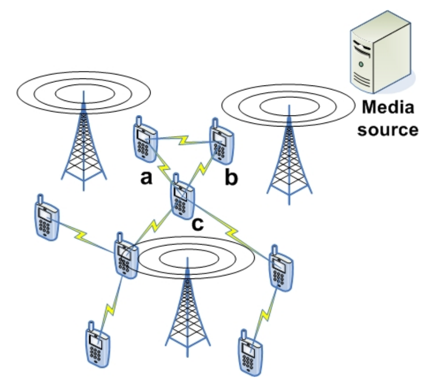


Fig. 1. Cooperative Peer-to-peer Repair (CPR) for WWAN Video broadcast. Peers a, b and c form a local repair group to share received WWAN packets.

One recent approach to alleviate the wireless packet loss problem is *cooperative communication* [3, 4]. In short, given the broadcast nature of wireless transmission (each transmission is heard by multiple receivers) and the "uncorrelatedness" of receivers' channels (hence unlikely for all receivers to undergo bad channel fades at the same time), peers that currently experience good channels (rich peers) can relay overheard information to clients who currently experience channel losses (poor peers), *without* relying on server retransmission. In the case of WWAN broadcast, if a secondary network such as an ad hoc Wireless Local Area Network (WLAN) connects the peers locally [3], a WWAN broadcast packet received by rich peers can be relayed to poor peers via ad hoc WLAN without interrupting primary WWAN transmission. We term this WLAN local recovery process *cooperative peer-to-peer repair* (CPR).

It has been shown [3] that instead of exchanging raw packets received directly from the WWAN streaming source (source packets), a peer can first encode a *structure network coding* (SNC) repair packet using received source packets before sharing the encoded repair packet with neighbors to further improve packet recovery. In details, a CPR packet of SNC type x is a weighted sum of received source packets belonging to frames in defined SNC group x in Galois Field $GF(2^F)$. Assuming the number of source packets in SNC group x is C_x, then a peer can recover all frames in SNC group x if *any* C_x innovative SNC packets of type $\leq x$ are received. Though SNC provides a powerful tool for peers to selectively recover frames in a CPR environment, the decisions of who should send repair packets of what SNC types at available transmission opportunities were not optimized in any formal way.

In this paper, we propose a distributed decision making strategy based on Markov decision process (MDP), so that each peer can make locally optimal transmission decisions based on observations eavesdropped on the WLAN channel. Our proposed MDP is both

978-1-61284-350-6/11/$26.00 ©2011 IEEE

computationally scalable and peer-adaptive, so that state transition probabilities in MDP can be estimated based on observed aggregate behavior of other peers. Experiments show that decisions made using our proposed MDP outperformed decisions made by a random scheme by at least 4dB in PSNR in received video quality.

The outline of the paper is as follows. We first discuss related work and overview our streaming system in Section 2 and 3, respectively. We then discuss the construction of MDP for CPR packet selection in Section 4. We discuss our experiments in Section 5 and conclude in Section 6.

2. RELATED WORK

Because the probability of packet loss in WWAN can be substantial, conventional WWAN video multicast schemes [2] involve a large overhead of FEC packets. Previous work on CPR [3] differs from these traditional approaches by utilizing a secondary ad hoc WLAN network for local recovery of packets lost in the primary network exploiting peers' cooperation. [3] has shown substantial gain in visual quality using CPR over non-cooperative schemes.

We stress that the assumption of multi-homing capable devices (each has multiple network interfaces to connect to orthogonal delivery networks simultaneously) is a common one in the literature [5, 6, 7], where different optimizations are performed exploiting the multi-homing property. [5] shows that aggregation of an ad hoc group's WWAN bandwidths can speed up individual peer's infrequent but bursty content download like web access. [6] proposes an integrated cellular and ad hoc multicast architecture where the cellular base station delivered packets to proxy devices with good channel conditions, and then proxy devices utilized local ad hoc WLAN to relay packets to other devices. [7] shows that smart striping of FEC-protected delay-constrained media packets across WWAN links can alleviate single-channel burst losses, while avoiding interleaving delay experienced in a typical single-channel FEC interleaver. [3] extended this body of multi-homed literature to WWAN video broadcast. We further improve [3] via formal optimization of the packet selection decision at each peer using distributed MDP.

MDP was first used for packet scheduling in a rate-distortion (RD) optimal manner in the seminal work *Rate-distortion optimized streaming* (RaDiO) [8]. Using partially observable MDP (POMDP), [9] extended the RaDiO work and considered scheduled transmissions of coded packets by a single sender using network coding when observations of client states are not perfect. [10] addressed the problem of selecting transmission video rates to neighboring peers in a P2P streaming scenario, where each peer must make its own decision in a distributed manner. Our work also leverages on the power of MDP in decision making; in particular, we tailor MDP to the WWAN video broadcast with CPR scenario, where the decision a peer must make is whether to transmit, and if so, which SNC type to encode a repair packet for local WLAN transmission.

3. SYSTEM OVERVIEW

We first overview components and operation of the previously proposed WWAN video broadcast system with CPR in Section 3.1. We then introduce our source and network models in Section 3.2. Finally, we discuss how network coding is used in CPR in Section 3.3.

3.1. WWAN Video Broadcast System with CPR

We consider a scenario where a group of N peers are watching the same WWAN broadcast video using their wireless multi-homed mobile devices. Each device is also equipped with a WLAN interface, and the peers are physically located in sufficiently close proximity that a peer-to-peer wireless ad hoc network can be formed. After each peer receives a potentially different subset of broadcast video packets through her WWAN interface (due to different network conditions experienced), they use their WLAN interfaces to exchange received WWAN packets to collectively recover packet losses in WWAN channels. We call this repair process *Cooperative Peer-to-Peer Repair* (CPR). See Fig. 1 for an illustration.

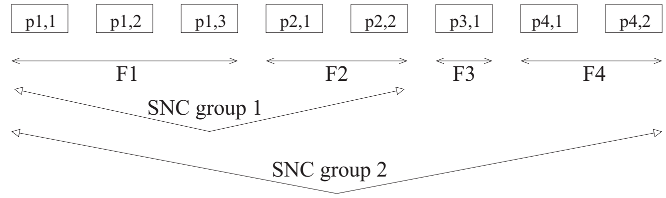


Fig. 2. Example of structure network coding (SNC) for a 4-frame GOP and two SNC groups: $\Theta_1 = \{F_1, F_2\}$, $\Theta_2 = \{F_1, \ldots, F_4\}$.

In more details, the operation of our WWAN video broadcast system with CPR can be explained in three phases. In the first phase, WWAN media source transmits an *epoch* worth of video of T seconds (a *group of picture* or GOP) to peers. In the second phase, peers exchange CPR packets via ad hoc WLAN to repair this GOP in time T during WWAN broadcast of the next GOP. (CPR repairs one GOP at a time.) In the third phase, after CPR completes its repair of a GOP in repair epoch of duration T, each peer decodes video from source packets and displays decoded video for consumption. Note that T is hence also the *repair epoch* in which CPR must complete its repair in a given GOP. The initial playback buffer delay for each peer is therefore two repair epochs. In practice, a GOP is on the order of 10 to 30 frames, hence at 15 frames per second, initial playback buffer delay of two repair epochs is only several seconds.

3.2. Source and Network Model

3.2.1. Source Model

We assume a GOP is composed of a leading I-frame followed by $K-1$ P-frames; each predicted frame F_k is differentially coded using the previous frame F_{k-1} as predictor. A frame F_k is correctly decoded if it is correctly received *and* its predictor (if any) is correctly decoded. A correctly decoded frame F_k reduces visual distortion by d_k. The objective is to maximize the expected distortion reduction of all CPR-participating peers. Each frame F_k is divided into r_k packets, $p_{k,1}, \ldots, p_{k,r_k}$, for transmission over packet-switched networks. The total number of source packets is then $R = \sum_{k=1}^{K} r_k$. Fig. 2 shows an example of a 4-frame GOP. F_1 has $r_1 = 3$ packets, F_2 and F_4 have $r_2 = r_4 = 2$ packets each, and F_3 has $r_3 = 1$ packet.

3.2.2. WWAN Channel Model

We model WWAN server-client transmission channel—the residual channel as seen from the application after lower-layer link adaptation have been applied—as a Gilbert-Elliott (GE) model, shown in Fig. 3, with parameters α and β for state transition probabilities between good (G) and bad (B) states. Within each state, packets containing compressed video frames are lost in an independent and identically distributed (iid) loss process with parameters a and b for G and B states, respectively, where $a < b$. We assume peers are physically located more than one transmission wavelength apart, and hence their channel statistics are (more or less) independent. Usually sufficient

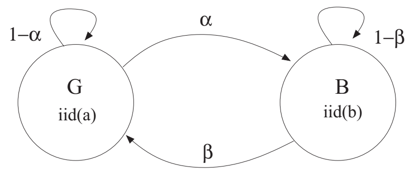


Fig. 3. Gilbert-Elliott loss model for WWAN broadcast channel. State transition probabilities from Good (G) to Bad (B) state and back are α, β, respectively. G and B state have iid loss process with loss rate a and b, respectively.

application-layer FEC are applied so that source packets are well protected when a client is in G state. Given the "cliff" effect of FEC, that means a client stuck in B state will experience significant source packet losses.

3.2.3. Ad hoc WLAN Channel Model

We assume packets are lost in the ad hoc WLAN due to in-air collision from hidden terminals. Denote by $\gamma_{n,m}$ the probability of a transmitted packet by peer n being lost to a one-hop receiving peer m. Loss probabilities $\gamma_{n,m}$'s are in general different for different pairs of peers. For simplicity, we assume they are known and unchanging for the duration of a repair epoch.

Packet transmissions in the ad hoc WLAN are scheduled according to the underlying 802.11 MAC layer protocol. Each peer contends for the right to send in the common tranmission medium via carrier sensing and exponential backoff [11]. Packets are sent/received in the promiscuous mode, which means a transmitted packet can be overhead by multiple neighboring peers within range.

When the right to send is granted by the MAC layer, a *transmission opportunity* (TO) becomes available to the peer. The peer then have to decide whether to send a packet during this TO, and if so, what packet to transmit. This is the question we seek to answer using distributed MDP in Section 4.

When a receiver correctly receives a CPR packet from a neighboring peer, it broadcasts an acknowledgment (ACK) packet. We assume that the ACK control packet, being very small compared to a data packet, is delivered without loss.

3.3. Network Coding for CPR

In order to improve CPR packet recovery efficiency, it has been proposed [3] that each peer should encode received WWAN packets into a coded packet using *network coding* (NC) [12] before performing CPR exchange. More specifically, at a particular TO for peer n, she has received set $\mathcal{G}_n$ of source packets from WWAN streaming source via WWAN and set $\mathcal{Q}_n$ of NC repair packets from neighboring peers via ad hoc WLAN. Peer n can NC-encode a CPR packet, q_n, as a randomized linear combination of packets in $\mathcal{G}_n$ and $\mathcal{Q}_n$:

$$q_n = \sum_{p_{i,j} \in \mathcal{G}_n} a_{i,j} p_{i,j} + \sum_{q_l \in \mathcal{Q}_n} b_l q_l \qquad (1)$$

where $a_{i,j}$'s and b_l's are random coefficients for the received source and CPR packets, respectively. This approach is called *Unstructured Network Coding* (UNC). The advantage of UNC is that *any* set of R received *innovative*[2] packets—resulting in R equations and R unknowns—can lead to full recovery of all packets in the GOP. The shortcoming of UNC is that if a peer receives fewer than R innovative packets, then this peer cannot recover *any* source packets using the received CPR packets.

[2]A new packet is innovative for a peer if it cannot be written as a linear combination of previously received packets by the peer.

3.3.1. Structure Network Coding for CPR

To address UNC's shortcoming, one can impose structure in the random coefficients $a_{i,j}$'s and b_l's in (1) when encoding a CPR packet, so that partial recovery of important frames in the GOP at a peer when fewer than R innovative packets are received is possible. Specifically, we define X *SNC groups*, $\Theta_1, \ldots, \Theta_X$, where each Θ_x covers a different subset of frames in the GOP and $\Theta_1 \subset \ldots \subset \Theta_X$. Θ_1 is the most important SNC group, followed by Θ_2, etc. Corresponding to each SNC group Θ_x is a *SNC packet type* x. Further, let $g(j)$ be index of the smallest SNC group that includes frame F_j.

As an example, in Fig. 2 frame F_1, F_2 are in SNC group Θ_1 and $F_1, \ldots, F_4$ are in SNC group Θ_2. The smallest SNC group that includes F_2 is Θ_1, i.e., $1 = g(2)$.

With the definitions above, a SNC packet $q_n(x)$ of type x can now be generated as follows:

$$q_n(x) = \sum_{p_{i,j} \in \mathcal{G}_n} \mathbf{1}(g(i) \leq x)\ a_{i,j} p_{i,j} + \sum_{q_l \in \mathcal{Q}_n} \mathbf{1}(\Phi(q_l) \leq x)\ b_l q_l, \qquad (2)$$

where $\Phi(q_l)$ returns the SNC type of packet q_l, and $\mathbf{1}(\mathtt{c})$ evaluates to 1 if clause $\mathtt{c}$ is true, and 0 otherwise. In words, (2) states that a CPR packet $q_n(x)$ of type x is a random linear combination of received source packets of frames in SNC group Θ_x and received CPR packets of type $\leq x$. Using (2) to generate CPR packets, a peer can now recover frames in SNC group Θ_x when $|\Theta_x| < R$ innovative packets of types $\leq x$ have been received.

More specifically, we can define the necessary condition[3] to NC-decode a SNC group Θ_x at a peer as follows. Let c_x be the sum of received source packets $p_{i,j}$'s such that $g(i) = x$, and received CPR packets of SNC type x. Let C_x be the number of source packets in SNC group x, i.e. $C_x = \sum_{F_k \in \Theta_x} r_k$. We can then define the number of *type x innovative packets* for SNC group Θ_x, I_x, recursively as follows:

$$\begin{aligned} I_1 &= \min(C_1, c_1) \\ I_x &= \min(C_x, c_x + I_{x-1}) \end{aligned} \qquad (3)$$

In words, (3) states that the number of type x innovative packets I_x is the smaller of C_x and the number of received SNC packets of type x, c_x, *plus* the number of type $x-1$ innovative packets in previous group I_{x-1}. A SNC group Θ_x is decodable if $I_x = C_x$.

4. DISTRIBUTED MARKOV DECISION PROCESS

We now address the packet selection problem when a peer is granted a TO by the MAC layer: should she send a CPR packet, and if so, which SNC type should the packet be NC-encoded in? We solve this problem via a carefully constructed MDP with finite horizon. We first present preliminaries discussion on how the MDP is set up in Section 4.1. We then discuss our chosen state & action space and state transition probabilities inside a MDP in Section 4.2 and 4.3, respectively. We discuss how the optimal decision policy is derived in Section 4.4. Finally, we discuss a procedure to reduce computation complexity of MDP in Section 4.5.

[3]This condition is not sufficient, however, since a received CPR packet can be a linear combination of other received packets. Given coefficients $a_{i,j}$'s and b_l's are chosen at random, this is expected to be a rare event.

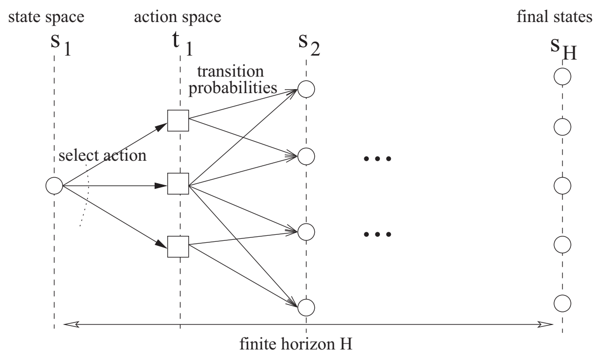


Fig. 4. Example of Markov Decision Process.

4.1. Preliminaries

We assume that at the start of the CPR repair epoch of T seconds, each peer n already knows who are her 1-hop neighbors, and who lost what source packets during WWAN transmission in the last epoch. Among her 1-hop neighbors, those who have lost at least one source packet in the last WWAN broadcast epoch are marked *target receivers*. At each MDP invocation (when a peer is granted a TO by the MAC layer), one peer m out of the pool of marked target receivers is selected (in a round robin fashion for fairness). The objective of the MDP is to maximize the expected distortion reduction of the selected target receiver m.

We assume the definition of SNC groups $\Theta_1, \ldots, \Theta_X$ are globally defined a priori as done in [3]; a peer just decides which SNC type to NC-encode a CPR packet given already defined SNC groups.

At each TO, a peer can estimate the number of TOs she has left until the end of the repair epoch based on the observed time intervals between previous consecutive TOs, and the amount of time remaining in the repair epoch. Let the estimated number of remaining TOs until the end of the repair epoch be H; hence H is also the finite horizon for the constructed MDP.

Finally, we assume that when a peer received a CPR packet from her neighbor, she immediately transmits a *rich* ACK packet, revealing her current state (the number of CPR packets she has received in each SNC type). Given ACK packets are always transmitted correctly (as discussed in Section 3.2.3), that means a sender always knows the state of its target receiver after a packet transmission.

4.2. State & Action Space for MDP

In a nutshell, a MDP of finite horizon H is a H-level deep recursion, each level t is marked by its states s_t's and actions a_t's. Each state s_t represents the state of the target receiver m $t-1$ TOs into the future, and a_t's are the set of feasible actions that can be taken by the sender at that TO given receiver's state s_t. The solution of a MDP is a *policy* π that maps each s_t to an action a_t, i.e., $\pi : s_t \mapsto a_t$.

We first define states s_t's for our MDP construction for SNC packet selection. Let $s_t(c_1, ..., c_X)$ be a feasible state for target receiver m at TO t, where c_x, as described in Section 3.3.1, is the sum of received CPR packets of SNC type x plus source packets of frame j such that $g(j) = x$. We assume c_x will not exceed C_x. The size of the state space is hence $O(\prod_{x=1}^{X} C_x)$, which can be large in practice whenever total number of source packets R is large, leading to a heavy computation burden when solving for the optimal decision policy via MDP. (Complexity reduction will be discussed later.)

For a feasible state $s_t(c_1, \ldots, c_X)$, we now define the corresponding action space. A CPR packet of SNC type x will not be transmitted if there are already sufficient number of packets to decode SNC group x. Thus, we declare an action $a_t = x$ to encode a CPR packet of SNC type x is feasible iff (if and only if) the following two conditions are satisfied:

1. There exists source packets $p_{i,j} \in \mathcal{G}_n$ such that $g(i) = x$ and/or $q_m \in \mathcal{Q}_n$ such that $\Phi(q_m) = x$.
2. $I_x < C_x$.

The first condition ensures there is new information pertaining to SNC group Θ_x that can be encoded, while the second condition ensures the encoded packet of type x is not unnecessary.

Note also that transmit nothing ($a_t = 0$) is always a feasible action.

4.3. State Transition Probabilities for MDP

Given defined states s_t's and action a_t's for MDP, we now derive state transition probabilities—the likelihood of landing in state s_{t+1} in next TO $t+1$ given state s_t and action a_t at current TO t. Here, we arrive at the "distributed" component of the packet selection problem: the probability of arriving in state s_{t+1} depends not only on the action a_t taken by this peer n at this TO t, but also actions taken by other peers transmitting to the target receiver m during the time between TO t and TO $t+1$. However, given packet selection is done by individual peers in a distributed manner (as opposed to centralized manner), how can this peer know what actions will be taken by other transmitting peers in the future?

Here, we leverage on previous work in distributed MDP [10] that utilizes the notion of users' *behavior patterns* in normal-form games [13]. The idea is to identify the patterns of users' tendencies to make decisions (rather than specific decisions), and then make prediction of users' future decisions based on these patterns. For our specific application of SNC type selection, we do the following. First, we assume that the number of received packets at target receiver m from other transmitting peer(s) between TO t and $t+1$ is L. This number can be learned from target receiver m's ACK messages between consecutive TOs.

Given L packets are received from other transmitting peer(s), we identify the corresponding SNC packet types by considering the following two aggregate behavior patterns. First pattern is *pessimistic* and assumes the aggregate of other transmitting peers always transmit innovative packets of the smallest SNC groups possible; i.e., packets of the smallest SNC group Θ_x such that $I_x < C_x$ will be transmitted until $I_x = C_x$, then the next smaller group Θ_{x+1} such that $I_{x+1} < C_{x+1}$, etc. This pattern is pessimistic because it seeks immediate benefit as quickly as possible, regardless of the number of TOs available in the finite horizon of H levels.

The second pattern is *optimistic* and assumes the aggregate of other transmitting peer(s) will always transmit innovative packets of the largest SNC group Θ_X. This is optimistic because it assumes all R type x innovative packets for the largest SNC group Θ_X will be received by the target receiver m, so that the entire GOP can be correctly decoded.

Let λ be the probability that a peer picks pessimistic pattern when selecting a SNC packet type. Then the probability that L packets are divided into k packets of pessimistic and $L-k$ packets of optimistic patterns is:

$$P(k|L) = \binom{L}{k} \lambda^k (1-\lambda)^{L-k} \tag{4}$$

In the absence of other information, we do not know which pattern is more likely, and we assume they are equally likely with probability $1/2$. The likelihood of each pattern will be learned from ACK messages from target receiver m as the CPR process progresses, however.

To derive the state transition probabilities, we first define G to be a mapping function that, given current state s_t, assigns SNC types for k pessimistic and $L-k$ optimistic packets into a corresponding SNC packet difference vector $\Delta = \{\delta_1, \ldots, \delta_X\}$, i.e. $G(s_t) : (k, L-k) \mapsto \Delta$. Let $\Delta^- = s_{t+1} - s_t$ be the group-by-group packet count difference between state s_{t+1} and s_t. Further, let $\Delta^+ = s_{t+1} - s_t - \{a_t\}$, which is like Δ^-, but also accounting for the CPR packet transmitted by this peer's current action a_t.

We can now write the state transition probability $P(s_{t+1}|s_t, a_t)$ of arriving in future state s_{t+1} given current state and action, s_t and a_t, as follows. Suppose the action taken $a_t = 0$ is not to transmit. Then we can write $P(s_{t+1}|s_t, a_t)$ as:

$$P(s_{t+1}|s_t, a_t = 0) = \sum_{k|(k,L-k) \overset{G(s_t)}{\to} \Delta^-} P(k|L) \tag{5}$$

In words, (5) is saying that to arrive at state s_{t+1}, the L packets received from other transmitting peers must be divided into pessimistic / optimistic packet counts $(k, L-k)$ that maps to packet difference vector Δ^-, i.e. the difference between s_{t+1} and s_t.

If $a_t = x$, i.e., peer chooses to transmit CPR packet of SNC type x, then

$$\begin{aligned} P(s_{t+1}|s_t, a_t = x) \;\; = \;\; & \gamma_{n,m} \sum_{k|(k,L-k) \overset{G(s_t)}{\longrightarrow} \Delta^-} P(k|L) \\ & +(1-\gamma_{n,m}) \sum_{k|(k,L-k) \overset{G(s_t)}{\longrightarrow} \Delta^+} P(k|L) \end{aligned} \tag{6}$$

In words, (6) is saying that to arrive at state s_{t+1}, the L packets received from other transmitting peers must lead to packet difference vector Δ^- if transmitted packet by this peer n is lost (with probability $\gamma_{n,m}$), or lead to packet difference vector Δ^+ if transmitted packet by this peer n is delivered successfully (with probability $1-\gamma_{n,m}$).

4.4. Finding Optimal Policy for MDP

The optimal policy π^* is one that leads to the minimum expected distortion (maximum distortion reduction) at the end of the H-level horizon. More specifically, denote by $\pi^*_t(s_t)$ the maximum expected distortion reduction at the end of H-level horizon, given state at TO t is s_t. $\pi^*_t(s_t)$ can be defined recursively: a chosen action a_t at TO t leads to state s_{t+1} with probability $P(s_{t+1}|s_t, a_t)$, and assuming optimal policy $\pi^*_{t+1}(s_{t+1})$ is performed at TO $t+1$, we have expected benefit $P(s_{t+1}|s_t, a_t)\pi^*_{t+1}(s_{t+1})$. $\pi^*_t(s_t)$ exhaustively searches for the optimal action a^*_t given state s_t:

$$\pi^*_t(s_t) = \begin{cases} \max_{a_t} \sum_{s_{t+1}} P(s_{t+1}|s_t, a_t)\pi^*_{t+1}(s_{t+1}) & \text{if } t < H \\ d(s_t) & \text{o.w.} \end{cases} \tag{7}$$

If state s_t is at the end of the H-level horizon, then no more actions can be taken and $\pi^*_t(s_t)$ in (7) simply returns the distortion reduction $d(s_t)$ given state s_t. $d(s_t)$ is defined as follows:

$$d(s_t) = \sum_{k=1}^{K} d_k \, \mathbf{1} \left(\bigcup_{x=g(k)}^{X} (I_x = C_x) \right) \tag{8}$$

(8) is basically saying that frame F_k can be recovered (to provide distortion reduction d_k) if any one of SNC groups $\Theta_{g(k)}, \ldots, \Theta_X$ has enough innovative packets for the SNC group to be correctly NC-decoded.

Note that (7) can be solved using *dynamic programming* (DP); i.e., each time (7) is solved, the optimal solution (maximum benefit and corresponding optimal action) is stored in DP table entry $\Pi[t][s_t]$, so that solution to a repeated subproblem can be simply looked up. The computation complexity of (7) is hence the number of computation per entry, $|a_t|$, times the number of entries, $H \times |s_t|$.

4.5. Reducing Complexity

At each TO, a peer constructs a MDP for a selected target receiver to determine the optimal SNC type to code a CPR packet. The complexity of finding the optimal policy π^* via (7) is determined by two factors: i) the depth of the recursion H, and ii) the size of the state space $|s_t|$. We can reduce the dimension of both factors as follows: instead of considering transmission and receipt of single packets, we consider multiples of M packets. That means each action $a_t = x$ implies transmission of CPR packet of SNC type x for M consecutive TOs, and finite horizon H/M means the number of multiple M back-to-back transmissions of CPR packets till the end of the repair epoch. Clearly, there will be occasions when transmission of a single packet of type x would suffice for correct decoding of SNC group Θ_x, hence this procedure does erode precision and the quality of the resulting solution. We will show in the experimental section that this is a worthwhile tradeoff, however.

5. EXPERIMENTATION

5.1. Experimental Setup

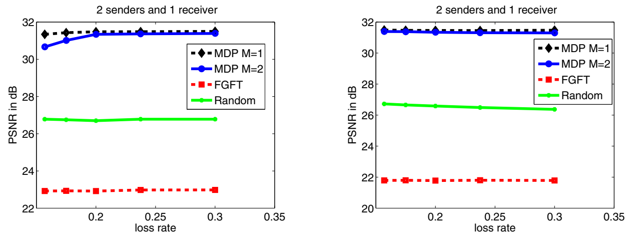
To test the performance of our proposed MDP, we set up the following experiment. For the video source, we used the video test sequence `akko` at 320×240 resolution encoded using H.264 JM version 16.2. In each GOP there were 10 frames: a leading I-frame plus 9 P-frames. We fixed the quantization parameters for I- and P-frames so that the resulting visual quality in Peak-Signal-to-noise Ratio (PSNR) was roughly 32.5dB. We assumed there are only two globally defined SNC groups: $\Theta_1 = \{F_1, \ldots, F_5\}$ and $\Theta_2 = \{F_1, \ldots, F_{10}\}$.

Maximum transport unit (MTU) were assumed to be 1250 bytes. For I- and P-frames, the average sizes were 5 and 1 packets, respectively. For WWAN broadcast, bandwidth was assumed to be 400kbps, and for each set of experiment, packet losses were simulated assuming GE model parameters α, β and a were fixed, while b was varied to effect loss rates from 0.15 to 0.3. For the ad hoc WLAN, we assumed that only a portion of the WLAN bandwidth equaled to 400kbps was available for CPR. WLAN loss rate was set to be 0.05 or 0.08.

5.2. Experimental Results

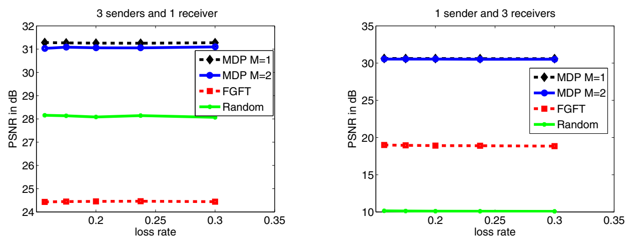
We compare the resulting video quality of our MDP-based packet selection (`MDP`) with two other schemes. In `FGFT`, each peer chooses to send CPR packets of SNC group Θ_1 first, and after sending sufficient number of type 1 CPR packets, she sends CPR packets of SNC group Θ_2. In `random`, each peer randomly chooses a packet type for transmission for each available TO.

In the first network topology, there were three peers. We focused on the case when exactly one peer (target receiver) entered B state in the GE model during a given WWAN transmission epoch, losing a substantial number of packets. The two remaining peers



(a) $\alpha = 0.01$, $a = 0.05$, $\beta = 0.8$ (b) $\alpha = 0.1$, $a = 0.04$, $\beta = 0.85$

Fig. 5. PSNR comparison of proposed MDP-based decision making and two other schemes for a three-node network topology.



(a) $\alpha = 0.01$, $a = 0.05$, $\beta = 0.8$ (b) $\alpha = 0.01$, $a = 0.05$, $\beta = 0.8$

Fig. 6. PSNR comparison of proposed MDP-based decision making and two other schemes for a four-node network topology.

(senders) remained in G state for the duration of the WWAN transmission epoch. In Fig. 5, we varied WWAN loss rate by varying loss rate b in B state (x-axis). y-axis is the resulting video quality in Peak Signal-to-Noise Ratio (PSNR) in dB. We can see that `MDP` outperformed `random` and `FGFT` by up to 5dB and 9dB, respectively. The main reason for the improvement is that `MDP` tried to maximize expected distortion reduction for the entire repair epoch, taking other transmitting peers' actions into consideration through observed behavior patterns. That means peers can avoid sending duplicate packets that would be non-innovative for the target receiver. We also observe that computation complexity of MDP can be reduced using reduction factor $M = 2$ (as discussed in Section 4.5), where the finite horizon H was at most 15, with a negligible reduction in performance.

In Fig. 5(b), we change the WWAN GE model parameters to be $\alpha = 0.1$, $a = 0.04$, $\beta = 0.85$. For the same network topology, WLAN loss rates for the two senders were set to be different values: 0.05 and 0.08. We see that `MDP` outperformed `random` and `FGFT` by 6dB and 9dB, respectively. In this case, one sender learned that the other sender had lower WLAN loss rate than herself, and subsequently elected not to transmit during available TOs for better overall system performance.

We have also experimented with an alternative network topology. Fig. 6(a) shows the performance comparison of the three schemes when there are four peers, with one target receiver (stuck in B state during WWAN broadcast) and three peers helping. In this topology, `MDP` outperformed `random` and `FGFT` by 4dB and 7dB, respectively. Finally, Fig. 6(b) is plotted for the scenario where there were four peers, with three target receivers and one peer helping. In this case, `MDP` outperformed `random` and `FGFT` by 11dB and 20dB, respectively.

6. CONCLUSION

In this paper, in a WWAN video broadcast scenario with cooperative peer-to-peer repair (CPR), where peers are listening to same video broadcast in the primary network while connected locally via a secondary network to share received packets, we propose a methodology for peers to make packet decisions in a distributed manner. In particular, at each transmission opportunity (TO) granted by the MAC layer, a peer finds the optimal policy via a constructed Markov decision process (MDP) to decide whether to forego transmission, or to select an SNC group to encode the CPR packet for transmission. To define the state transition probabilities inside the MDP, one needs to estimate the SNC types of CPR packets that will be transmitted by other peers to the target receiver between two TOs. We propose to use two behavior patterns, pessimistic and optimistic, to model the decision tendencies of other transmitting peers; the likelihood of each pattern can be learned via the target receiver's ACK messages. We also propose a simple procedure to reduce the computational complexity of the MDP at a modest cost of solution quality. Our experiments show that packet selection decisions made using our MDP-based strategy outperformed competing schemes by at least 4dB in decoded video quality.

7. REFERENCES

[1] J. Crowcroft and K. Paliwoda, "A multicast transport protocol," in *ACM SIGCOMM*, New York, NY, August 1988.

[2] 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.

[3] X. Liu, G. Cheung, and C.-N. Chuah, "Structured network coding and cooperative wireless ad-hoc peer-to-peer repair for WWAN video broadcast," in *IEEE Transactions on Multimedia*, 2009, vol. 11, no.4.

[4] O. Alay, P. Liu, Y. Wang, E. Erkip, and S. Panwar, "Error resilient video multicast using randomized distributed space-time codes," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Dallas, TX, March 2010.

[5] P. Sharma, S.-J. Lee, J. Brassil, and K. Shin, "Distributed communication paradigm for wireless community networks," in *IEEE International Conference on Communications*, Seoul, Korea, May 2005.

[6] R. Bhatia, L. Li, H. Luo, and R. Ramjee, "ICAM: Integrated cellular and ad hoc multicast," in *IEEE Transactions on Mobile Computing*, August 2006, vol. 5, no. 8, pp. 1004–1015.

[7] G. Cheung, P. Sharma, and S.J. Lee, "Smart media striping over multiple burst-loss channels," in *IEEE Transactions on Selected Topics in Signal Processing*, August 2007, vol. 1, no.2, pp. 319–333.

[8] 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.

[9] D. Nguyen and T. Nguyen, "Network coding-based wireless media transmission using POMDP," in *Seventeenth International Packet Video Workshop*, Seattle, WA, May 2009.

[10] H. Park and M. van der Schaar, "A framework for foresighted resource reciprocation in P2P networks," in *IEEE Transactions on Multimedia*, January 2009, vol. 11, no.1, pp. 101–116.

[11] J. Kurose and K. Ross, Eds., *Computer Networking: A Top-Down Approach (5th Edition)*, Addison Wesley, 2009.

[12] T. Ho, M. Medard, R. Koetter, D. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," in *IEEE Transactions Infomation Theory*, Oct. 2006, vol. 52, pp. 4413–4430.

[13] E. Haruvy, D. Stahl, and P. Wilson, "Evidence for optimistic and pessimistic behavior in normal-form games," in *Econ. Lett.*, 1996, vol. 63, pp. 255–259.