Delay-Optimized Network Coding for Video Streaming over Wireless Networks

Hulya Seferoglu, Athina Markopoulou
University of California, Irvine
{hseferog, athina}@uci.edu

***Abstract*—In this paper, we study delay-optimized network coding for video streaming over wireless networks with network coding capabilities. It has been demonstrated that network coding can increase throughput in wireless networks, by mixing packets from different flows into a single packet, thus increasing the information content per transmission. However, network coding potential is not always fully exploited in this setting, because there may not be enough packets at intermediate nodes to do network coding due to bursty nature of transport protocols, packet losses, and difference in path delays. One way to deal with this problem is to delay packets at intermediate nodes in order to create more network coding opportunities. However, introducing large or varying delays eventually hurts video traffic, which requires low delay and delay jitter. In this paper, we study this tradeoff in coded wireless networks, and we propose a packet delaying scheme at intermediate nodes to maximize video quality.**

I. INTRODUCTION

Providing high quality video over wireless networks is a challenging problem, due to both the erratic and time-varying nature of wireless channels and the stringent delivery requirements of video traffic. Developments in video compression and streaming, wireless networking, and cross-layer design, are continuously advancing the state-of-the art in wireless video [1], [2]. Network coding has emerged from the pioneering work in [3], [4], and it has been shown that it improves throughput. Wireless networks naturally lend themselves to network coding, thanks to the inherent broadcast and overhearing capabilities. We are particularly interested in video streaming over wireless mesh networks with opportunistic network coding, which have been extensively studied in theory and practice [5], [6].

In this setting, network coding promises significant gains, but there are several issues in practice. One of the issues is the rate mismatch between flows, which can significantly reduce network coding opportunities, as there may not be enough packets from different flows at intermediate nodes to code together due to bursty nature of transport protocols, packet losses, and difference in path delays. One possible solution to this problem is to delay packets at intermediate nodes. A packet delaying scheme is proposed in [7], and it is shown to be optimal in terms of throughput and video quality improvement for one-hop wireless networks. However, in the context of more general network topologies, such as multi-hop wireless mesh networks, the throughput improvement and video quality achieved by the proposed scheme are far from the optimal. The authors in [8] propose to delay packets to improve the performance of TCP over coded wireless mesh networks, and they observe that the throughput increases with increasing delay, but only up to a certain extent; excessive delay reduces the TCP rate, thus decreasing the overall throughput. The optimal delay depends on the network topology and the background traffic and also may change over time. In this paper, we propose a packet delaying scheme at intermediate nodes by taking into account topology and traffic related parameters to maximize video quality.

This paper builds on our recent work in [9] where we formulated video-aware network coding schemes in rate-distortion optimization (RaDiO) framework over multi-hop wireless mesh networks with one-hop opportunistic network coding of unicast flows (as implemented in COPE [6]). The proposed schemes improve network throughput and video quality by taking into account network coding and video specific properties. In this paper, first, we formulate the problem of delay optimization at intermediate nodes in RaDiO framework and propose a distributed solution. Our solution takes into account network coding opportunities and their evolution in time as well as video specific properties, such as distortion values and deadlines of video packets. Second, we design a practical mechanism; *video-aware packet delaying* using the structure of the optimal solution. Finally, we compare our video-aware packet delaying scheme to the baselines without packet delaying and/or network coding.

The rest of paper is organized as follows. Section II gives an overview of the system model. Section III presents the proposed delay-optimized network coding scheme and its practical implementation. Section IV presents simulation results that demonstrate the benefits of the proposed scheme over baselines, in terms of video quality. Section V concludes the paper.

II. SYSTEM OVERVIEW

Coded wireless networks. We consider wireless mesh networks where intermediate nodes (wireless mesh routers) are able to forward packets to other intermediate nodes and/or clients. We assume that intermediate nodes can perform simple network coding operations (XOR) and combine packets from several incoming flows into a single outgoing packet. This packet is broadcast to the entire neighborhood, thus reaching several nodes at the same time. We assume that nodes can overhear all transmissions in their neighborhood, whether they are intended for them or not; they can decode a network-coded packet using overheard packets. The idea of combining network coding with broadcast to increase the information content per transmission is well understood in the network coding community. This idea has been applied in 802.11-based multi-hop wireless networks and throughput benefits have been demonstrated for data applications [6].

Sources. We assume that a set of video flows are transmitted, between some known source-destination pairs over a wireless mesh network with network coding capabilities. Nodes make network coding and delay decisions by taking into account the importance and deadline requirements of video packets. The distortion value (Δ) of every packet can be determined by the video source and communicated to the intermediate nodes in order to enable them to make decisions about transmission of video packets in a rate-distortion optimized manner [10]. This information can be marked on a special field of the packet header. This field can be at the application level (*e.g.*, RTP extended headers) or a part of the network coding header; alternatively, the typically unused TOS/DiffServ byte in the IP header can be overridden. In addition to the individual importance of packets (Δ) within a flow, our formulation also considers the importance of flows (γ). In general, the overall importance of a packet can be a function of the flow priority and the packet distortion value; in this paper, we use a simple product $\gamma \cdot \Delta$.

Routing. In this paper, we consider that each video flow follows a single path from the source to the destination. Paths are pre-determined by a routing protocol and known a-priori.

Video-aware packet delaying for network coding. Our focus is on the formulation of delay-optimized network coding and on the development of video-aware packet delaying schemes in wireless mesh networks.

Example. The example shown in Fig. 1 demonstrates the key intuition why we need to delay packets. There are two flows in reverse directions: node A transmits a video flow to node B, and node B transmits a video flow to node A via relay node I. All nodes transmit over the same channel and at the same power level. The link capacities are inversely proportional to the distance between the node pairs. In this example, B has already transmitted its packet; b_1 to the relay. When a new transmission opportunity arises, all three nodes can content to transmit their packets; A; a_1, B; b_2; and I; b_1. It is clear that there is no network coding opportunity in this example. However, if I does not transmit b_1, even if it captures the channel, and waits to receive a_1, it can combine a_1 and b_1, and reduce the number of transmissions (thus increase the throughput). A trivial solution is to make I wait till it has a network coding opportunity. However, this increases the end-to-end delay of video flows. The unpredictable delay at I also increases delay jitter which is not welcome for video applications. As a result, a video-aware scheme that arranges the amount of delay is needed to create more network coding opportunities. □

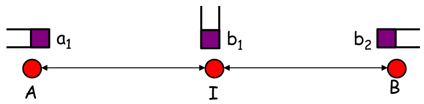

Fig. 1. Alice(A) & Bob(B) topology. A transmits a video packet a_1 to B, and B transmits a video packet b_1 to A over I. I is the intermediate node which combines packets from A and B, and broadcasts.

III. DELAY OPTIMIZATION

1) Formulation: Let us consider a single node $n \in \mathbf{N}$ in a wireless mesh network, with packets $\mathbf{\Phi}_n = \{p_1, p_2, ..., p_{\phi_n}\}$ in its queue where $\phi_n = |\mathbf{\Phi}_n|$. Without network coding and packet delay, in order to do classic RaDiO packet scheduling, the node would choose a transmission policy $\boldsymbol{\pi}$ for the next transmission opportunity. The transmission policy would indicate for every packet in the queue, $p_j \in \mathbf{\Phi}_n$, whether this packet is transmitted $\pi(j) = 1$ or not $\pi(j) = 0$, so as to minimize a weighted function of distortion and rate $J(\boldsymbol{\pi}, \lambda) = D(\boldsymbol{\pi}) + \lambda R(\boldsymbol{\pi})$.

With network coding and code delay, the node $n \in \mathbf{N}$ chooses some network codes, consisting of packets in the queue XOR-ed together, to transmit. All possible network codes at node n are elements of the network coding set; $\boldsymbol{C}_n$. The code policy $\mathbf{\Theta}_n$ at node n consists of transmission policy vector $\mathbf{\Pi}_n = \{\Pi_n(\boldsymbol{c}_u) : \forall \boldsymbol{c}_u \in \boldsymbol{C}_n\}$ and delay policy $\mathbf{\Omega}_n = \{\Omega_n(\boldsymbol{c}_u) : \forall \boldsymbol{c}_u \in \boldsymbol{C}_n\}$ where $\boldsymbol{c}_u$ can be considered as both network code and a set of network coded packets. The code policy $\mathbf{\Theta}_n$ is expressed as $\mathbf{\Theta}_n = \{\Pi_n(\boldsymbol{c}_u), \Omega_n(\boldsymbol{c}_u) : \forall \boldsymbol{c}_u \in \boldsymbol{C}_n\}$. The transmission policy $\mathbf{\Pi}_n$ indicates for every possible code $\boldsymbol{c}_u \in \boldsymbol{C}_n$, whether it is transmitted $\Pi_n(\boldsymbol{c}_u) = 1$ or not $\Pi_n(\boldsymbol{c}_u) = 0$, in the next transmission opportunity. The delay policy $\mathbf{\Omega}_n$ indicates for every possible code $\boldsymbol{c}_u \in \boldsymbol{C}_n$, the amount of its delay (in terms of number of transmission opportunities or number of slots) before transmission if it is not to be transmitted in the next transmission opportunity. To avoid transmitting two network codes $\boldsymbol{c}_u, \boldsymbol{c}_v \in \boldsymbol{C}_n$ that have common packets from the set $\mathbf{\Phi}_n$, we restrict our attention to "valid" network code transmission policies $\mathbf{\Pi}_n^{valid} \subset \mathbf{\Pi}_n$ with $\Pi_n^{valid}(\boldsymbol{c}_u) = 1 \bigwedge \Pi_n^{valid}(\boldsymbol{c}_v) = 1$ if and only if $\boldsymbol{c}_u \bigcap \boldsymbol{c}_v = \emptyset$. Then, the valid network code policy is defined as $\mathbf{\Theta}_n^{valid} = \{\Pi_n^{valid}(\boldsymbol{c}_u), \Omega_n(\boldsymbol{c}_u) : \forall \boldsymbol{c}_u \in \boldsymbol{C}_n\}$.

Our goal is to find the optimal code policy on all nodes $\mathbf{\Theta}^{valid} = \{\mathbf{\Theta}_n^{valid} : \forall n \in \mathbf{N}\}$, so as to minimize the total distortion $D(\mathbf{\Theta}^{valid})$ subject to the rate constraint $R(\mathbf{\Theta}^{valid}) \leq R_{avg}$, where R_{avg} is the available bit rate. With Lagrangian relaxation, our problem is equivalent to finding the code policy $\mathbf{\Theta}^{valid}$ that minimizes the Lagrange function $J(\mathbf{\Theta}^{valid}, \lambda) = D(\mathbf{\Theta}^{valid}) + \lambda R(\mathbf{\Theta}^{valid})$.

Instead of finding the optimal code policy, we can map each code to the packets it contains (*i.e.*, packets that are XOR-ed together), and find the optimal packet policy, packet transmission and delay policies. The reason for converting the problem from a code to a packet policy selection is that it is more natural to express distortion values per packet. Let $\boldsymbol{\pi}$

be the packet transmission policy on all nodes, $\boldsymbol{\pi} = \{\pi_n(j) : \forall n \in \mathbf{N}, \forall p_j \in \boldsymbol{\Phi}_n\}$. $\boldsymbol{\pi}$ depends on the code transmission policy $\boldsymbol{\Pi}^{valid}$ as follows:

$$\pi_n(j) = \begin{cases} 1 & \text{if } \exists\ \boldsymbol{c}_u \in \boldsymbol{C}_n \text{ s.t. } p_j \in \boldsymbol{c}_u \wedge \Pi_n^{valid}(\boldsymbol{c}_u) = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Let $\boldsymbol{\omega}$ be the packet delay policy on all nodes, $\boldsymbol{\omega} = \{\omega_n(j) : \forall n \in \mathbf{N}, \forall p_j \in \boldsymbol{\Phi}_n\}$. $\omega_n(j)$ depends on the code transmission policy $\boldsymbol{\Pi}^{valid}$ and code delay policy $\boldsymbol{\Omega}$ as follows:

$$\omega_n(j) = \begin{cases} \Omega_n(\boldsymbol{c}_u) & \text{if } \pi_n(j) = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

An equivalent problem is to optimize the packet policy; $\boldsymbol{\theta} = \{\boldsymbol{\theta}_n : \forall n \in \mathbf{N}\}$ where $\boldsymbol{\theta}_n = \{\pi_n(j), \omega_n(j) : \forall p_j \in \boldsymbol{\Phi}_n\}$ and $\theta_n(j) = \{\pi_n(j), \omega_n(j)\}$. The optimization problem;

$$\min_{\boldsymbol{\theta},\lambda} J(\boldsymbol{\theta}, \lambda) = \min_{\boldsymbol{\theta},\lambda} \{D(\boldsymbol{\theta}) + \lambda R(\boldsymbol{\theta})\} \tag{3}$$

where $D(\boldsymbol{\theta})$ is the total distortion over all nodes, and $R(\boldsymbol{\theta})$ is the the total rate function over all nodes under policy $\boldsymbol{\theta}$. Let us explain $D(\boldsymbol{\theta})$ and $R(\boldsymbol{\theta})$ in detail.

$D(\boldsymbol{\theta})$ is the approximate distortion of the flows transmitted from node n under policy $\boldsymbol{\theta}$: $D(\boldsymbol{\theta}) = \sum_{\forall n \in \mathbf{N}} D(\boldsymbol{\theta}_n)$. Following a similar definition as in [11], $D(\boldsymbol{\theta}_n) = \sum_{p_j \in \boldsymbol{\Phi}_n} \gamma(j)\Delta(j)P(\theta_n(j))$, where: $\gamma(j)$ is the priority/importance of the flow to which packet p_j belongs; $\Delta(j)$ is video quality distortion when packet p_j is lost as defined in II; and $P(\theta_n(j))$ is the probability that packet p_j is lost under policy $\theta_n(j)$. In particular, $P(\theta_n(j)) = P_p(\theta_n(j))P_c(\theta_n(j))$ consists of two parts: the probability $P_p(\theta_n(j))$ that the packet is lost in previous transmissions till its $\omega_n(j)$ delayed transmission under policy $\theta_n(j)$; and the probability $P_c(\theta_n(j))$ that the packet is lost in its $\omega_n(j)$ delayed transmission under policy $\theta_n(j)$. The packet loss probabilities; $P_p(\theta_n(j))$ and $P_c(\theta_n(j))$ can be further expressed as follows:

$$\begin{aligned} P_p(\theta_n(j)) = \prod_{m=1}^{M} \{ & P\{ftt_n > t_d(j) - t_m | rtt_n > t_t(j) - t_m\} \\ & P\{rtt_n > t_t(j) - t_m\}\}, \end{aligned}$$

$$P_c(\theta_n(j)) = \begin{cases} P\{ftt_n > t_d(j) - t_t(j)\} & \text{if } \pi_n(j) = 1, \\ 1 & \text{if } \pi_n(j) = 0 \end{cases} \tag{4}$$

where M is the number of transmissions of the packet p_j till current time t_c, ftt_n is the random variable of forward trip time from node n to the receiver, rtt_n is the random variable of round trip time from node n to the receiver and back to the node n, $t_d(j)$ is the deadline of the packet p_j, t_m is the m^{th} transmission time of packet p_j, and $t_t(j)$ is the delayed transmission time of packet p_j. The delayed transmission time $t_t(j)$ is: $t_t(j) = t_c + \kappa_n \omega_n(j)$ where t_c is the current time and κ_n is the average duration of slots. Since $\omega_n(j)$ is defined as the delay of packet p_j in terms of slots, $t_t(j) = t_c + \kappa_n \omega_n(j)$ is the delayed transmission time of packet p_j. Similarly to [10], we assume that ftt_n and rtt_n follow a gamma distribution each, assuming that we know average delay at each hop.

$R(\boldsymbol{\theta})$ is the total rate function over all nodes under policy $\boldsymbol{\theta}$: $R(\boldsymbol{\theta}) = \sum_{\forall n \in \mathbf{N}} R(\boldsymbol{\theta}_n)$. $R(\boldsymbol{\theta}_n)$ is the rate of the flows transmitted from node n under policy $\boldsymbol{\theta}_n$: $R(\boldsymbol{\theta}_n) = B \sum_{p_j \in \boldsymbol{\Phi}_n} \rho(\theta_n(j)) z_n(\theta_n(j))$ where B is the size of packet $p_j \in \boldsymbol{\Phi}_n$ in bytes, $\rho(\theta_n(j))$ is the average cost of transmitting packet p_j, and $z_n(\theta_n(j))$ is the rate improvement factor of p_j and represented as $z_n(\theta_n(j)) = \frac{1}{|\boldsymbol{c}_u(\theta_n(j))|}$. $\boldsymbol{c}_u(\theta_n(j))$ is the expected network code set of the current code set $\boldsymbol{c}_u$ in the future under policy $\theta_n(j)$. Note that $\boldsymbol{c}_u(\theta_n(j)) = \boldsymbol{c}_u$ when the packet is decided to be transmitted in the current transmission opportunity (*i.e.*, $\pi_n(j) = 1 \wedge \omega_n(j) = 0$). Otherwise, $\boldsymbol{c}_u(\theta_n(j))$ depends on the topology, packet loss probabilities over wireless channels, and path delays. Its value is estimated over time by using a learning algorithm or it is determined analytically if underlying topology is known. We analytically determine $\boldsymbol{c}_u(\theta_n(j))$ in our evaluations.

By replacing the distortion $D(\boldsymbol{\theta})$ and rate $R(\boldsymbol{\theta})$ terms with their detailed expressions discussed above and by noting that $\sum_{p_j \in \boldsymbol{\Phi}_n}$ and $\sum_{\forall \boldsymbol{c}_u \in \boldsymbol{C}_n} \sum_{\forall p_j \in \boldsymbol{c}_u}$ are equivalent, the problem of Eq. (3) is expressed as follows:

$$\begin{aligned} J(\boldsymbol{\theta}, \lambda) = & \sum_{n \in \boldsymbol{N}} \sum_{\forall \boldsymbol{c}_u \in \boldsymbol{C}_n} \sum_{p_j \in \boldsymbol{c}_u} (\gamma(j)\Delta(j)P(\theta_n(j)) + \\ & \lambda B \rho(\theta_n(j)) z_n(\theta_n(j))) \end{aligned} \tag{5}$$

2) Optimal Solution: Since current systems typically transmit one packet (network code in our case) at each transmission opportunity, we will focus on this case from now on. Let us define $J_n^{\boldsymbol{c}_u}(\boldsymbol{\theta}_n, \lambda) = \sum_{p_j \in \boldsymbol{c}_u} (\gamma(j)\Delta(j)P(\theta_n(j)) + \lambda B \rho(\theta_n(j)) z_n(\theta_n(j)))$ where $J(\boldsymbol{\theta}, \lambda) = \sum_{n \in \boldsymbol{N}} \sum_{\forall \boldsymbol{c}_u \in \boldsymbol{C}_n} J_n^{\boldsymbol{c}_u}(\boldsymbol{\theta}_n, \lambda)$. Now we consider to transmit network a network code $\boldsymbol{c}_u \in \boldsymbol{C}_n$. To make a transmission decision, considering Eq. (1) and Eq. (2), $J_n^{\boldsymbol{c}_u}(\{1, \omega_n(j)\}, \lambda) \leq J_n^{\boldsymbol{c}_u}(\{0, \omega_n(j)\}, \lambda)$ should be satisfied, [10], [9]. The Lagrange multipliers that satisfy this inequality;

$$\begin{aligned} & \lambda_n^{\boldsymbol{c}_u}(\Omega_n(\boldsymbol{c}_u)) = \\ & \frac{\sum_{p_j \in \boldsymbol{c}_u} \gamma(j)\Delta(j)(P_p(\{0,0\}) - P(\{1, \Omega_n(\boldsymbol{c}_u)\}))}{B \sum_{p_j \in \boldsymbol{c}_u} z_n(\{1, \Omega_n(\boldsymbol{c}_u)\})}, \end{aligned} \tag{6}$$

where $\lambda_n^{\boldsymbol{c}_u}(\Omega_n(\boldsymbol{c}_u))$ is calculated $\forall n \in \boldsymbol{N}, \boldsymbol{c}_u \in \boldsymbol{C}_n$ and for all $\Omega_n(\boldsymbol{c}_u)$ values. $\max_{\{n, \boldsymbol{c}_u, \Omega_n(\boldsymbol{c}_u)\}} \lambda_n^{\boldsymbol{c}_u}(\Omega_n(\boldsymbol{c}_u))$ gives the optimal transmission and delay policy.

3) Video-Aware Packet Delaying: The optimal transmission and delay policies are found by maximizing the Lagrange multipliers in Eq. (6); $\max_{\{n, \boldsymbol{c}_u, \Omega_n(\boldsymbol{c}_u)\}} \lambda_n^{\boldsymbol{c}_u}(\Omega_n(\boldsymbol{c}_u))$. There are two drawbacks of the optimal scheme; (i) high computational complexity and (ii) high overhead (to exchange Lagrange multipliers among all nodes). Therefore, we propose a suboptimal scheme to the optimal solution; *Video-Aware Packet Delaying*. In this scheme, each node $n \in \boldsymbol{N}$ maximizes the Lagrange multiplier in Eq. (6) for $\Omega_n(\boldsymbol{c}_u) = 0$. The optimal network code is c_u^*. For $c_u = c_u^*$, Eq. (6) is maximized again to determine whether it is beneficial to delay this network code for the optimal delay $\Omega_n^*(\boldsymbol{c}_u^*) \neq 0$. If $\Omega_n^*(\boldsymbol{c}_u^*) \neq 0$ exists, the node *needs delay*.

A node keeps both $\lambda_n^{\boldsymbol{c}_u^*}(0)$ and $\lambda_n^{\boldsymbol{c}_u^*}(\Omega_n^*(\boldsymbol{c}_u^*))$ values, and advertises them to its neighbors (the advertisement is done through control packets of the underlying MAC layer). When a transmission opportunity arises, for node n to make a transmission decision, there are two cases whether the node (i) needs delay or (ii) not. For the first case, node n compares its $\lambda_n^{\boldsymbol{c}_u^*}(0)$ to its neighbor's Lagrange multipliers. If it has maximum value and all other nodes either do not have any packets to transmit or need delay, the node transmits its network code. In the second case, the node again compares its $\lambda_n^{\boldsymbol{c}_u^*}(0)$ to its neighbor's Lagrange multipliers. If it has the largest value among all its neighbors, then it transmits. Otherwise, it only transmits if all its neighbors need delay.

IV. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed scheme; video-aware packet delaying (which we call *NC-RaDiO-Delay* in this section) in terms of video quality in different scenarios. We compare it to two baseline schemes: *NC-RaDiO* and *noNC*. We observe that video-aware packet delaying scheme significantly improves the video quality over baseline schemes.

We used the GloMoSim simulation environment [12] to implement the proposed algorithm and the baseline schemes. We consider a cross topology where multiple flows cross at an intermediate node as shown in Fig. 2: pairs of nodes A, B and C, D communicate over an intermediate node I, *e.g.*, A transmits to B and B transmits to A via I. We consider this topology to evaluate the proposed schemes, because it is a simple topology and representative, and it brings out the main points. A single channel is used for both uplink and downlink transmissions and channel capacity of each link is the same and varies from $150kbps$ to $1Mbps$ in our simulations. In this scenario, the intermediate node I makes decisions on network coding and packet delaying. We assume that nodes are placed on a circle with center I and radius $90m$. We consider a generalized version of the cross topology for different number of nodes; $N : 3-9$ including the intermediate node. Packets may be lost due to errors on the wireless channel. We consider the two-ray path loss model and Rayleigh fading channel model implemented in GloMoSim. The two-ray path loss model is a propagation path loss model using free space path loss for near sight and plane earth path loss for far sight. For the Rayleigh fading model, we consider average channel SNR $\{3, 5, 7, 9\}$ dB in our simulations. Packets may also experience a random MAC propagation delay, $2ms$ on average. IEEE 802.11 is used in the MAC layer, with similar modifications proposed in [6] needed for network coding. The one-way delay budget for each flow is set to $150ms$ in the simulations.

As our test sequences, we used standard sequences: *Carphone*, *Foreman*, *Mother & Daughter*, *Claire*, *Coastguard*, *News*, *Grandma*, and *Salesman*. These were QCIF sequences encoded using the JM 8.6 version of the H.264/AVC codec [13], [14]. The group of pictures consisted of one I and nine P frames. All encoded sequences had data rate $70kbps$ and frame rate $30fps$. Each frame consists of at least one slice. Each slice was packetized into an independent NAL (network abstraction layer) unit of size $250B$. NAL units are encapsulated using the Real-time Transport Protocol (RTP) and User Datagram Protocol (UDP). As metric for the video quality of an encoded sequence, we use the average PSNR, *i.e.*, the peak-signal-to-noise ratio based on the luminance (Y) component of video sequences, measured in dB, and averaged over the entire duration of the video sequence. The PSNR of the encoded sequences *Carphone, Foreman* and *Mother & Daughter*, before any transmission, was $29.95dB$, $28.70dB$ and $40.74dB$ respectively. We repeated and concatenated the standard sequences to create longer test sequences of duration $30sec$ each. At the receiver side, basic copy-concealment scheme is used when an entire frame is lost.

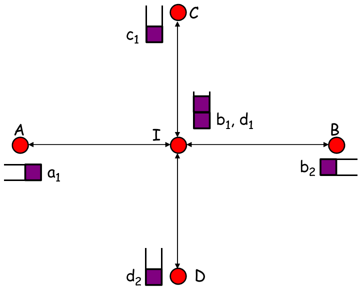

Fig. 2. Cross topology. A, B and C, D transmit video flows to each other over the intermediate node I. I combines packets opportunistically and broadcasts.

We compare our algorithm, NC-RaDiO-Delay to two baseline algorithms: noNC, NC-RaDiO. noNC is a FIFO queue without network coding. In every time slot, I transmits the first packet from the head of the queue. NC-RaDiO is a rate-distortion optimized network code design and scheduling scheme developed for video streaming in [9]. This scheme does not delay packets in intermediate nodes to create more network coding opportunities.

Fig. 3 presents the average video quality (PSNR) versus channel capacity for the cross topology in Fig. 2 when $N = 3$. The delay budget is $150ms$ and the channel SNR is set to $11dB$. The figure shows that NC-RaDiO-Delay and NC-RaDiO outperform noNC all the time. It is expected since these schemes are getting advantages of both network coding and video-aware packet scheduling. NC-RaDiO-Delay improves over NC-RaDiO up to $3dB$ and it improves over noNC up to $4dB$. This shows that packet delaying increases network coding opportunities, hence the video quality. All three algorithms have similar performances for high channel capacities. This is expected because at high channel capacities even noNC scheme is able to transmit all the data and achieves maximum achievable video quality.

Fig. 4 shows the simulation results for different channel SNR levels from $3dB$ to $11dB$ for the cross topology in Fig. 2 when $N = 3$, delay budget is $150ms$ and the channel

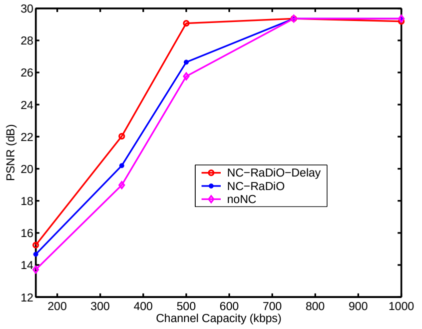

Fig. 3. Video quality (PSNR) versus channel capacity for *cross topology* with $N = 3$ nodes including I. Delay budget is $150ms$, and channel SNR is $11dB$.

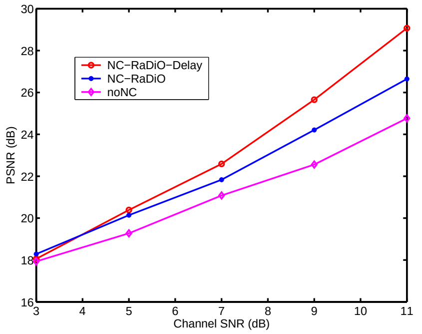

Fig. 4. Video quality (PSNR) versus channel quality (SNR) for *cross topology* with $N = 3$ nodes including I. Delay budget is $150ms$, and channel capacity is $500kbps$.

capacity is $500kbps$. The results show that NC-RaDiO-Delay improves over NC-RaDiO up to $2.5dB$ and noNC up to $5dB$. The improvement of NC-RaDiO-Delay over NC-RaDiO and the improvement of NC-RaDiO over noNC are lower for low channel SNR values and higher for high SNR values. The reason is that at low channel SNR values, a considerable number of the packets are lost and network coding opportunities are limited even if packets are delayed.

Fig. 5 shows the simulation results for different number of nodes (from $N = 3$ to $N = 9$) for the cross topology in Fig. 2 when the delay budget is $150ms$, the channel capacity is $1Mbps$, and the channel SNR level is $11dB$. The results show that NC-RaDiO-Delay improves over NC-RaDiO up to $2dB$ and noNC up to $6dB$. It is seen that all three algorithms have similar performance for $N = 3$, because there are plenty of opportunities to transmit packets even if there is no network coding. The improvement becomes clear when the number of nodes increases (*i.e.*, per flow channel capacity decreases). All simulations in this section conclude that packet delaying creates more network coding opportunities and improves the quality of network coded video flows.

V. CONCLUSION

In this paper, we proposed an approach to improve video quality over wireless mesh networks with opportunistic network coding. The key intuition was to fully exploit network coding potential over coded networks by eliminating the rate mismatch between flows that are coded together, through delaying packets at intermediate nodes. First, we formulated delay-optimized network coding scheme by taking into account the importance of packets and the evolution of network codes in time to determine network codes to transmit and packet scheduling. Second, we developed an efficient video-aware packet delaying scheme considering the structure of the optimal solution. The simulation results show that the proposed scheme improves video quality up to $3dB$ and $6dB$ compared to baseline schemes.

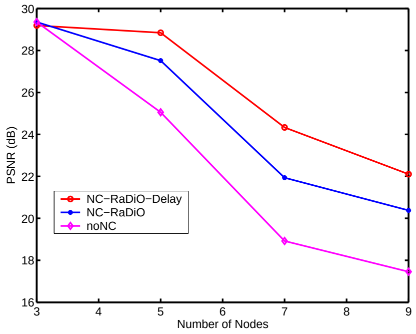

Fig. 5. Video quality (PSNR) versus the number of nodes for *cross topology* (from $N = 3$ to $N = 9$ including I). Delay budget is $150ms$, channel capacity is $1Mbps$, and channel SNR level is $11dB$.

REFERENCES

[1] N. Färber and B. Girod, "Wireless Video," in A. Reibman, M.-T. Sun (eds.), *Compressed Video over Networks*, M. Dekker, 2000.

[2] Special issue on "Advances in Wireless Video," *in IEEE Wireless Communications Magazine*, Aug. 2005.

[3] R. Ahlswede, N. Cai, S. R. Li, and R. W. Yeung, "Network information flow," *in IEEE Trans. on Inf. Theory*, vol. 46(4), pp. 1204-1216, July 2000.

[4] S. R. Li, R. W. Yeung, and N. Cai,"Linear network coding," *in IEEE Trans. on Information Theory*, vol. 49(2), pp. 371-381, Feb. 2003.

[5] Y. Wu, P. A. Chou, and S. Y. Kung, "Information exchange in wireless network coding and physical layer broadcast," *in Proceedings of IEEE CISS*, Baltimore, MD, March 2005.

[6] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: Practical network coding," *in IEEE/ACM Trans. on Networking*, vol. 16(3), pp. 497-510, June 2008.

[7] D. Nguyen, T. Nguyen, X. Yang, "Multimedia Wireless Transmission with Network Coding," *in Proc. of the* 16^{th} *Packet Video Workshop*, Lausanne, Switzerland, Nov. 2007.

[8] Y. Huang, M. Ghaderi, D. Towsley, W. Gong, "TCP Performance in Coded Wireless Mesh Networks," *in Proc. of IEEE SECON*, San Francisco, CA, June 2008.

[9] H. Seferoglu, A. Markopoulou, "Video-Aware Opportunistic Network Coding over Wireless Networks", *in JSAC, Special Issue on Network Coding over Wireless Communication Networks,* vol. 27(5), June 2009.

[10] P. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," *in IEEE Trans. on Multimedia*, vol. 8(2), pp. 390- 404, Apr. 2006.

[11] J. Chakareski and P. Frossard, "Rate-Distortion Optimized Distributed Packet Scheduling of Multiple Video streams over Shared Communication Resources," *in IEEE Trans. on Multimedia*, vol. 8(2), pp. 207-218, 2006.

[12] GloMoSim Version 2.0 , "Global Mobile Information Systems Simulation Library," *available at http://pcl.cs.ucla.edu/projects/glomosim/.*

[13] ITU-T Rec. "H.264 : Advanced video coding for generic audiovisual services", 2005.

[14] H.264/AVC Reference Software Version JM 8.6 *http://iphome.hhi.de/suehring/tml/download/*, 2006.