Frame Corruption Estimation from Route Messages for Video Coding over Mobile Ad Hoc Networks

Yiting Liao and Jerry D. Gibson
Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA, 93106
Email: {yiting, gibson}@ece.ucsb.edu

***Abstract*—Recently we proposed a cross-layer design to support video communication over error-prone mobile ad-hoc networks. The idea is to utilize routing messages and network parameters to estimate the corrupted frames, and to guide the reference frame selection at the video encoder to mitigate error propagation. In this paper, we focus on the frame corruption estimation method used in the design. We build a packet loss probability model from the MAC layer mechanism and network parameters; then we utilize the model along with the routing messages received at the network layer to estimate the possible corrupted frames. We study our estimation method under different network settings and demonstrate its effectiveness and robustness. We further show the video quality gains achieved by adapting reference frame selection to the estimated results.**

I. Introduction

Mobile ad-hoc networks have drawn increasing attention in recent years due to their wide deployment in military, homeland defense, and disaster recovery applications. Supporting real-time services such as voice and video over such networks is essential but not easily accomplished due to the dynamic nature of the network. Node mobility and the lack of infrastructure in the network leads to frequent link failures and route changes while the effects of fading, noise and interference in the wireless medium may affect the link quality.

For video transmitted over such lossy networks, a major problem is that error propagation introduced in the motion-compensated prediction loop may greatly degrade the delivered video quality. There are many research efforts to address this problem, including intra refresh techniques [1], [2], redundant picture coding [3], and feedback-based reference picture selection (RPS) [4]. Intra refresh techniques insert intra macroblocks (MBs) or pictures to minimize the effect of error propagation, while redundant picture coding methods allocate redundant pictures to increase the delivered video quality. However, these techniques introduce certain reductions in coding efficiency. On the other hand, the coding efficiency penalty of video coding with RPS is much lower and results in [5] show that the RPS algorithms provide better error resilience than conventional intra refresh techniques. However, RPS requires extra overhead to transmit control messages between the encoder and the decoder, which may lead to higher network traffic load and increased delay.

This research has been supported by the UC Discovery Grant Program and Applied Signal Technology, Inc.

In our previous work [6], we proposed to use the standard ad-hoc routing messages to estimate the possible packet losses in the networks and select reference frames accordingly to alleviate error propagation caused by packet losses. It has been shown to be an effective method to enhance error resilience of video without introducing extra network overhead or delay. In [6], we use a simple method to utilize the routing messages, that is, every time a route error (RERR) message is received by the source node, we assume that the previously transmitted packet is lost. However, due to the transmission delay of the video packets and routing messages, a RERR may indicate possible losses of several previously transmitted video packets.

In this paper, we establish a new model to estimate the packet loss probability of preceding packets transmitted from the source node before a RERR is received. This model utilizes information from MAC access mechanism and network parameters for the packet loss estimation. We further estimate the corrupted frames using the estimation model and routing messages received from the network layer. We run experiments using the Qualnet simulator to test the effectiveness of our design and we show that our estimation method can effectively detect the frame corruption in the network while maintaining a low probability of false alarm. We further demonstrate that our estimation model works well under various network settings.

The estimated results can be adapted to the reference frame selection approach for multiple description coding with multipath transport [7] (referred to as RA-MDC) and help to alleviate the effect of error propagation. In this paper, we show that the RA-MDC achieves PSNR gains of up to 2.16 dB under different packet loss rates. In [7], we discuss the RA-MDC method in more detail and provide more comprehensive results on delivered video quality for multiple users.

II. Frame Corruption Estimation

In this section, we present our frame corruption estimation method. First, we build a packet loss probability model for the preceding packets sent from the source node when a route error message is received at the source node. Based on the model and the routing messages, we then estimate the packet loss probability of each transmitted packet at the source node and determine whether the corresponding frame is corrupted or not.

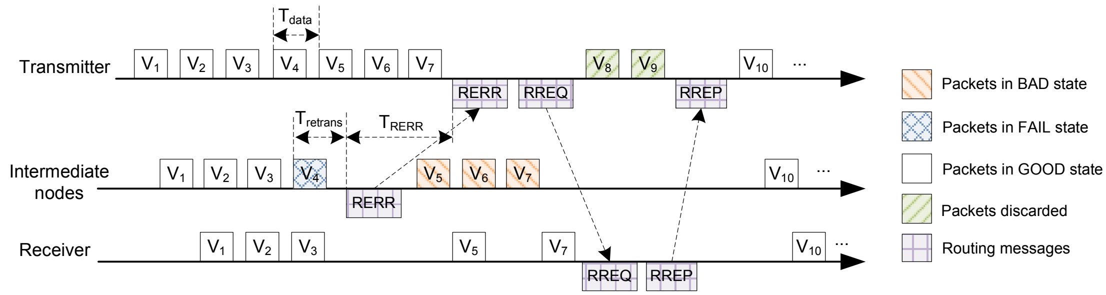

Fig. 1. An example to illustrate the packet losses in the network and the corresponding routing messages

A. Packet Loss Probability Model

For most on-demand routing protocols over ad-hoc networks, a route error (RERR) message is initiated when the MAC layer fails all retransmission attempts to transmit a packet to the next hop destination. This RERR indicates that a link becomes unreliable and packets transmitted through this link suffer a high packet loss rate. Before the source node receives the RERR, video packets sent from the source node are still transmitted through this error-prone link and are susceptible to losses. When the source receives the RERR, it either reconstructs the route from the route cache or initiates a route recovery process to find a new route. A new route is established when the route reply (RREP) message reaches the source node. Packets scheduled to be transmitted in the broken route during the route recovery process are discarded and marked as lost. According to the routing mechanism, when the source node receives a RERR message, it indicates a link becomes unreliable and packets previously sent through this link suffer packet losses. We derive a model to estimate the packet loss probability of the preceding packets sent through this unreliable link.

Figure 1 illustrates how the RERR message correlates to the packet losses in the networks. As shown in Fig. 1, a RERR is initiated at the intermediate node when video packet v_4 exhausts all retransmission attempts and still fails to transmit to the next hop destination. We define the retransmission delay of this packet as T_{retrans}. After time T_{RERR}, the source node receives the RERR and stops transmitting video packets through the unreliable link. We see that packets v_5, v_6, v_7 sent during time period $T_{\text{retrans}}+T_{\text{RERR}}$ are still transmitted through the unreliable link and are very susceptible to packet loss.

We assume that anytime the source receives a RERR, the preceding video packets sent from the source node follow the same packet loss distribution under the same network conditions. Therefore, we denote $\text{Pr}(n)$ as the packet loss probability of the n^{th} preceding packet sent from the source node before the source node receives a RERR. Our main goal is to model $\text{Pr}(n)$ and utilize it to determine the potential corrupted frames. Due to the random delay between link failure and RERR reception at the source, the n^{th} preceding packet before RERR can be sent at a time before, right at or after the link failure happens. We use three states to represent these three cases: GOOD means the packet is sent before the link failure, FAIL means the packet fails to transmit and triggers RERR, and BAD means the packet is sent after the link failure. According to our above analysis, we define $\text{Pr}(n)$ as

$$\text{Pr}(n) = \lambda_g \cdot p_g(n) + \lambda_f \cdot p_f(n) + \lambda_b \cdot p_b(n) \qquad (1)$$

where λ_g, λ_f, and λ_b represents the packet loss probability in GOOD, FAIL, or BAD state respectively, and $p_g(n)$, $p_f(n)$, and $p_b(n)$ denotes the probability of the n^{th} preceding packet in these three states, respectively. In the following, we estimate the state probability distribution and packet loss probabilities in these three states.

1) Estimation of State Probability Distribution: The state of a video packet depends on the delay of the link failure feedback and the transmission interval of video packets. For example, as shown in Fig. 1, v_4 is the packet that triggers RERR and hence is in FAIL state. The packets sent before v_4 (e.g. v_3) are in GOOD state while the packets sent after v_4 are in BAD state. Therefore, we can compare the video packet transmission interval T_{data} and the delay of the link failure feedback T_{delay} to determine the state probability of the packets sent before receiving the RERR by

$$\begin{cases} p_g(n) = p\left(T_{\text{delay}} \le (n-1)T_{\text{data}}\right) \\ p_f(n) = p\left(nT_{\text{data}} \ge T_{\text{delay}} > (n-1)T_{\text{data}}\right) \\ p_b(n) = p\left(T_{\text{delay}} > nT_{\text{data}}\right) \end{cases} \qquad (2)$$

We can calculate the video packet interval T_{data} by

$$T_{\text{data}} = L/R_t \qquad (3)$$

where R_t is the transmission bitrate of the video sequence and L is the payload size. Then in order to calculate Eq. (2), we need to estimate the probability distribution of T_{delay}.

As shown in Fig. 1, we see that T_{delay} consists of two parts: the retransmission delay of a packet that fails all retransmission attempts (denoted as T_{retrans}) and the time period to transmit the RERR to the source (denoted as T_{RERR}). So we have

$$T_{\text{delay}} = T_{\text{retrans}} + T_{\text{RERR}} \qquad (4)$$

The values of both T_{retrans} and T_{RERR} depend on the MAC layer access mechanism. In this paper, our estimation is based on the 802.11 DCF basic access mechanism [8].

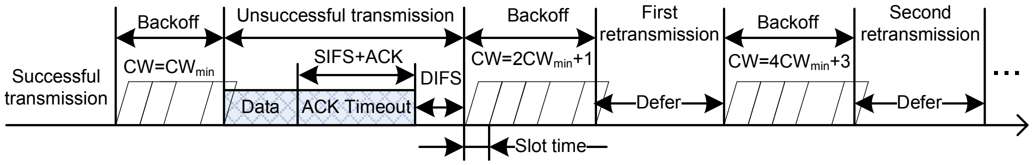

Fig. 2. Packet retransmission procedure based on basic access mechanism

a) $T_{\rm RERR}$: We first estimate the time period to transmit the RERR to the source $T_{\rm RERR}$ by

$$T_{\rm RERR} = n_{\rm hop} \cdot T_C \tag{5}$$

where $n_{\rm hop}$ is the average number of hops to transmit RERR to the source, T_C is the transmission time for a successful RERR transmission defined in [8].

b) $T_{\rm retrans}$: We then estimate the transmission delay $T_{\rm retrans}$ of a packet that fails to transmit from the current station to the next hop destination after exhausting all retransmission attempts. As shown in Fig. 2, each transmission period consists of a defer access and a backoff process. The transmission procedure starts when the station senses an idle channel and invokes a backoff procedure. The backoff time is uniformly chosen in the range of $[0, CW]$, where *CW* is the current contention window (CW) size. Then the station sends out the video packet. If the transmitting station does not receive the ACK, the station concludes that the transmission has failed and invokes a retransmission process until the retransmission limit is reached. Note that CW takes an initial value of $CW_{\min}$ and exponentially increases after each unsuccessful transmission, until it reaches the maximum CW size of $CW_{\max}$.

Based on the above analysis, the transmission delay of a packet that fails all retransmission attempts is

$$T_{\rm retrans} = mT_D + T_{\rm backoff} \tag{6}$$

where m is the retransmission limit, T_D is the time period of a defer access defined in [8], and $T_{\rm backoff}$ is the overall backoff time.

The overall backoff time is a random variable that is the sum of a series of independent random variables uniformly distributed in the range of $[0, W_i] \cdot T_{\rm slot}$ and W_i is the *CW* size in the i^{th} retransmission defined in [8].

We define T_{B_i} as the backoff time in the i^{th} retransmission, then we have $T_{B_i} \sim U(0, W_i \cdot T_{\rm slot})$, where $U(0, W_i \cdot T_{\rm slot})$ represents an uniform distribution in the range $[0, W_i] \cdot T_{\rm slot}$. Thus the overall backoff time is

$$T_{\rm backoff} = \sum_{i=0}^{m-1} T_{B_i} \sim U_{\rm s}(0, \sum_{i=0}^{m-1} W_i \cdot T_{\rm slot}) \tag{7}$$

where $U_{\rm s}(\cdot)$ represents the probability distribution of the overall backoff time $T_{\rm backoff}$, which is the sum of m uniform random variables. We use $P_s(t)$ to represent the CDF of $T_{\rm backoff}$, *i.e.* the probability that $T_{\rm backoff}$ is shorter than time t is represented by $P_s(t)$.

Finally, based on Eqs. (2)-(7), we have the state probability distribution by

$$\begin{cases} p_g(i) = P_s(\Delta T) \\ p_f(i) = P_s(T_{\rm data} + \Delta T) - P_s(\Delta T) \\ p_b(i) = 1 - P_s(T_{\rm data} + \Delta T) \end{cases} \tag{8}$$

where $\Delta T = (n-1)T_{\rm data} - T_{\rm RERR} - mT_D$.

2) Estimation of Packet Loss Probability λ_g, λ_f, and λ_b: λ_g refers to the packet loss rate of a good link, in which the ACK is received to indicate a successful transmission. Therefore, we assume $\lambda_g = 0$. λ_b is defined as the packet loss rate of an unreliable link, which is the probability that the video packet does not reach the next hop destination successfully. λ_f is the packet loss rate for the video packet that fails all transmission attempts and triggers the RERR. Based on the MAC layer mechanism, we know that each time a video packet fails a transmission, it means either the video packet fails to transmit to the next hop destination or the ACK message is not received by the transmitter. Thus, λ_f is the conditional packet loss probability for the video packet that fails all transmission attempts.

Let A_0 denote the event that the video packet is lost and A_1 denote the event that the video packet fails all transmission attempts. We assume that each transmission is independent and the loss probability of a video packet and an ACK for an unreliable link are $p_{\rm data}$ and $p_{\rm ACK}$ respectively. Then we have $p(A_0) = p_{\rm data}^m$ and $p(A_1) = [p_{\rm data} + (1 - p_{\rm data}) \cdot p_{\rm ACK}]^m$. Finally, λ_f and λ_b are represented by

$$\begin{aligned} \lambda_f &= p(A_0|A_1) = \frac{p(A_1|A_0) \cdot p(A_0)}{p(A_1)} \\ &= \frac{p_{\rm data}^m}{[p_{\rm data} + (1 - p_{\rm data}) \cdot p_{\rm ACK}]^m} \end{aligned} \tag{9}$$

$$\lambda_{\rm b} = p(A_0) = p_{\rm data}^m \tag{10}$$

By Eqs. (9) and (10), we have

$$\frac{\lambda_b}{\lambda_f} = [p_{\rm data} + (1 - p_{\rm data}) \cdot p_{\rm ACK}]^m \leq 1 \tag{11}$$

i.e. λ_f is generally larger than λ_b.

B. Frame Corruption Estimation

Section II-A presents a packet loss probability model for the preceding packets sent from the source node when a RERR message is received. Every time the source node receives a RERR message, we estimate the packet loss probability of the preceding packets using the proposed estimation model. For the packets that are dropped during the route recovery process,

TABLE I
SIMULATION PARAMETERS

Region	500 m× 500 m
Number of nodes	50
Mobility model	Random waypoint model: node speed 0 ∼ 10 m/s, pause time 120 s
PHY data rate	5.5 Mbps
Transmission Power	15 dBm
MAC layer protocol	802.11b CSMA/CA
Playout deadline	350 ms

we set their packet loss probabilities to 1. Given the estimated packet loss probability of each video packet, we can estimate the frame corruption probability by

$$p(f_k) = 1 - \prod_{\{v_i|v_i \in f_k\}} (1 - p(v_i)) \tag{12}$$

where $p(v_i)$ is the packet loss probability of packet v_i, $p(f_k)$ is the frame corruption probability of frame f_k, and $\{v_i|v_i \in f_k\}$ is the set of packets that contain information of frame f_k.

We define a threshold p_{thres} to determine whether a frame is corrupted or not, i.e. if $p(f_k) \geq p_{\text{thres}}$, we consider frame f_k as corrupted. We propose to use the frame corruption estimation results to assist the reference frame selection in video coding. That is, we do not use any estimated corrupted frame as a reference frame in the motion compensation process in order to mitigate error propagation.

III. EXPERIMENTAL RESULTS

To investigate the effectiveness and robustness of our estimation model, we simulated a two-path transport system over a mobile ad-hoc network using the Qualnet simulator. The network parameters chosen are shown in Table I. In this ad-hoc network, 50 nodes are uniformly placed in a 500m×500m region, where the connectivity of any two nodes is determined by the network topology and the transmission power. The movement of each node is characterized by a random waypoint model [9] with parameters shown in Table I. We use IEEE 802.11b with 5.5 Mbps PHY transmission rate and CSMA/CA basic access protocol. The values of IEEE 802.11b parameters used for the packet loss probability model can be found in [8]. In the network layer, we implement the split multipath routing (SMR) protocol [10] to generate two routes for multipath transport. A pair of source and destination nodes is randomly chosen to transmit video packets and packets are dropped if they do not reach the destination by the playout deadline of 350 ms.

We use two error probabilities to measure the accuracy of our estimation. We define a false alarm probability P_{FA} as the probability of detecting a corrupted frame when the frame is actually correctly received and define a miss-detection probability P_{MISS} as the probability of detecting a correctly received frame when the frame is actually corrupted.

We plot a receiver operating curve (ROC) [11] to represent the possible values of P_{FA} and P_{MISS} under various p_{thres} in the range of [0, 1] in Fig. 3. This figure is generated

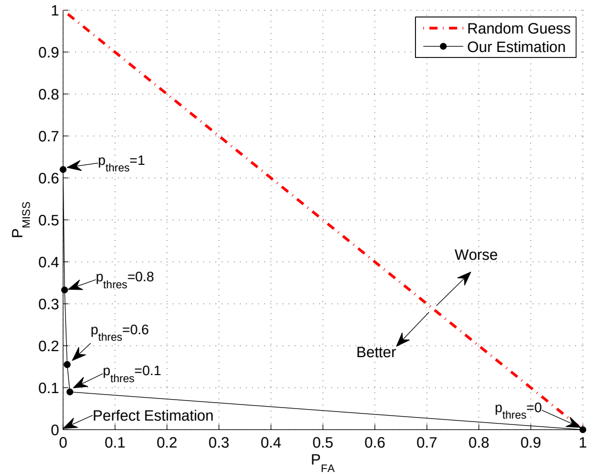

Fig. 3. ROC curve for $p_{\text{thres}} \in [0, 1]$. Varying p_{thres} provides a trade-off between P_{FA} and P_{MISS}.

TABLE II
P_{FA}, P_{MISS} UNDER DIFFERENT TRANSMISSION POWERS WITH $p_{\text{thres}} = 0.5$

Transmission Power	P_{FA}	P_{MISS}
15 dBm	0.008	0.155
14 dBm	0.010	0.195
13 dBm	0.012	0.229
12 dBm	0.014	0.267
11 dBm	0.016	0.315

using the default network settings. In this ROC space, the (0, 0) point represents perfect estimation and the diagonal line denotes a completely random guess. The overall accuracy of the estimation depends on how close the point is to the lower left corner. In Fig. 3, we see that the ROC curve of our estimation method is close to the lower left corner for the p_{thres} values in the range of [0.1, 0.6], where we achieve a very low probability of a false classification while maintaining a fairly low probability of missing a corrupted frame. Thus, by choosing a p_{thres} value in that range, our estimation method yields fairly good performance.

Next, we examine the robustness of our estimation method under different network settings. Here we choose $p_{\text{thres}} = 0.5$. Table II presents P_{FA} and P_{MISS} values for transmission power varied in the range of 11∼15 dBm. We see that the false alarm probability is constantly low under these network settings, which leads to negligible unnecessary reduction in coding efficiency. Meanwhile, the miss-detection probability is in the range of 0.16∼0.32, which indicates that our proposed estimation method can detect most of the corrupted frames. We notice that the performance of the estimation becomes worse as the transmission power is reduced. This is because when the transmission power decreases, the network connectivity becomes worse. The loss probability of routing messages may increase, which leads to more miss-detections.

Table III shows the impact of the number of nodes on our estimation accuracy. As shown in Table III, the estimation accuracy does not have distinguishable difference

TABLE III
P_{FA} AND P_{MISS} UNDER DIFFERENT NUMBER OF NODES WITH $p_{\mathrm{thres}} = 0.5$

Number of Nodes	P_{FA}	P_{MISS}
20	0.008	0.144
40	0.008	0.164
50	0.008	0.155
60	0.008	0.169
80	0.008	0.164

under a different number of nodes, which means that the estimation accuracy is insensitive to the number of nodes in the network. Similarly, we study estimation accuracy under different network sizes, transmission bitrates, and number of retransmissions. In general, the false-alarm probability is lower than 1% while the miss-detection probability is lower than 32%. Therefore, our estimation model is effective and robust under various network settings.

Finally, we evaluate the delivered video quality while our frame corruption estimation is used for video coding. We incorporate our frame corruption estimation with the reference frame selection technique for multiple description video coding with multipath transport (MPT) [7]. We refer to this method as routing-aware multiple description coding (RA-MDC). We compare our proposed method to single description coding (SDC) and multiple description coding (MDC) with MPT. We implement multiple state video coding (MSVC) [12] as the MDC method and apply refined error concealment method as proposed in [13]. For the three methods, we use the same MPT strategy such that even and odd frames are transported through two routes respectively.

In the simulation shown here, we use foreman sequence (CIF) at 15 fps with 150 frames and the bitrate is 400 kbps. We vary the transmission power from 15 dBm to 11 dBm to achieve packet loss rates in the range of $2.2\% \sim 8.8\%$. For each packet loss rate, we simulate 500 realizations and plot the average PSNR in Fig. 4. We see that our RA-MDC method achieves up to 1.44 dB gain in PSNR compared to MDC and up to 2.16 dB gain compared to SDC, which shows that our proposed RA-MDC can effectively mitigate error propagation and improve the delivered video quality. More details on the RA-MDC method can be found in [7].

IV. Conclusions

In this paper, we propose a frame corruption estimation method for video coding over mobile ad hoc networks. We notice that the routing mechanisms in most of the routing protocols provides feedback concerning the network conditions. This inspires us to utilize these feedback messages to estimate the packet losses in the network and to adapt the video coding accordingly. We build a packet loss probability model from the network parameters and estimate the frame corruption probability based on the estimation model and the routing messages received at the network layer. We demonstrate that our estimation method is effective and robust under various network settings. In addition, we use the estimated results to guide the reference frame selection for multiple description

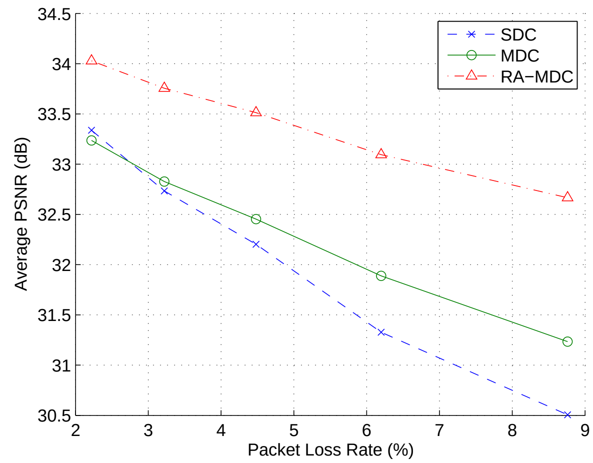

Fig. 4. Performance under different packet loss rates for Foreman sequence (CIF, 15fps) at 400 kbps. Transmission power varies from 15 dBm to 11 dBm to achieve packet loss rate from 2.2% to 8.8%.

video coding and show that our method achieves up to 2.16 dB gains in PSNR.

References

[1] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 966 –976, Jun 2000.

[2] T. Stockhammer, D. Kontopodis, and T. Wieg, "Rate-distortion optimization for JVT/H.26L video coding in packet loss environment," in *International Packet Video Workshop*, 2002.

[3] C. Zhu, Y.-K. Wang, M. M. Hannuksela, and H. Li, "Error resilient video coding using redundant pictures," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 1, pp. 3–14, Jan. 2009.

[4] Y.-K. Wang, C. Zhu, and H. Li, "Error resilient video coding using flexible reference frames," in *Proceedings of SPIE*, vol. 5960, no. 1. SPIE, 2005.

[5] Y. Wang, A. R. Reibman, and S. Lin, "Multiple description coding for video delivery," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 57–70, Jan. 2005.

[6] Y. Liao and J. D. Gibson, "Routing-aware multiple description video coding over wireless ad-hoc networks using multiple paths," in *International Conference on Image Processing (ICIP)*, Sep. 2010.

[7] ——, "Routing-aware multiple description coding with multipath transport for video delivered over mobile ad-hoc networks," accepted by IEEE Workshop on Multimedia Communications & Services, 2010.

[8] "IEEE Standard Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999)*, pp. C1–1184, 2007.

[9] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, 1998, pp. 85–97.

[10] S. J. Lee and M. Gerla, "Split multipath routing with maximally disjoint paths in ad hocnetworks," in *IEEE International Conference on Communications*, vol. 10, 2001, pp. 3201–3205.

[11] R. Yates and D. Goodman, *Probability and Stochastic Processes: A Friendly Introduction for Electrical and Computer Engineers*. John Wiley & Sons, 2004.

[12] J. G. Apostolopoulos, "Reliable video communication over lossy packet networks using multiple state encoding and path diversity," in *SPIE Proceedings on Visual Communications and Image Processing*, vol. 4310, no. 1, 2001, pp. 392–409.

[13] Y. Liao and J. D. Gibson, "Refined error concealment for multiple state video coding over ad hoc networks," in *Proceedings of the 42nd Asilomar Conference on Signals, Systems and Computers*, Oct. 2008, pp. 2243 –2247.