

Hindawi Publishing Corporation
Research Letters in Communications
Volume 2008, Article ID 546184, 4 pages
doi:10.1155/2008/546184

Research Letter

An Analytical Model for Optimum Byte-Level and Packet-Level FEC Assignment Using Buffer Dynamics

Azfar Moid and Abraham O. Fapojuwo

Department of Electrical and Computer Engineering, University of Calgary, Calgary, AB, Canada T2N 1N4

Correspondence should be addressed to A. Moid, amoid@ucalgary.ca

Received 8 January 2008; Accepted 18 March 2008

Recommended by Luca De Nardis

This letter presents an analytical model that jointly exploits the buffer dynamics of both the sending and receiving nodes to find the optimum number of byte-level and packet-level forward error correction (FEC) units for real-time multimedia transmission over wireless networks. The proposed analytical model first provides an optimum number of FEC units required at the byte-level, and then chooses the number of FEC units at the packet-level based on current channel and network conditions. The accuracy of the proposed model is dependent on two parameters: the variable deadline-time at the byte-level and fixed round-trip time (RTT) delay at the packet-level. Numerical results demonstrate the effectiveness of the model in reducing the unrecoverable error probability, which is achieved when the byte-level FEC scheme is supplemented by the packet-level FEC scheme.

Copyright © 2008 A. Moid and A. O. Fapojuwo. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. Introduction

In this letter, the problem of selecting the byte-level and packet-level forward error correction (FEC) combinations that increase the overall system efficiency and video quality is presented as a 0-1 knapsack optimization problem. We propose a dynamic programming (DP) based algorithm for selecting the byte-level and packet-level FEC units to minimize the effect of error-prone wireless channel and network losses. This approach is different from that of [1] by calculating the cost/throughput ratio (CTR) at both the byte and packet levels, instead of a single level considered in [1]. Moreover, unlike in [1] where the time parameter is kept constant, here we adopt a methodology that incorporates the variable time parameter in calculating CTR, which is a more realistic approach. The work presented in this letter also supplements the work in [2] by presenting an analytical model that proves the effectiveness of using packet-level FEC in conjunction with byte-level FEC schemes. The main contribution of this letter is the presentation of a DP-based model that utilizes the queuing dynamics of both the sending and receiving nodes of a wireless network in calculating the number of FEC redundancy units at both the byte and packet

levels, in the context of video transmission over a wireless network.

2. Problem Formulation and Proposed Solution

Suppose that m_b and M_p are the particular FEC combinations chosen at the byte-level and packet-level, respectively, such that $m_b = (n, k_b)$ and $M_p = (N_p, K)$. Here, n is the total number of bytes per packet after redundancy is added to a packet and k_b is the original number of bytes per packet, such that $n = k_b + b$, and b ($0 \leq b \leq n - 1$) is the number of redundancy bytes added to a packet. Similarly, N_p is the total number of packets in a video block after redundancy is added and K represents the original number of packets in a video block, such that $N_p = K + p$, and p ($0 \leq p \leq K$) is the number of redundant packets added to a video block.

2.1. The optimization problem

Formally, the optimization problem is casted as a 0-1 knapsack problem, and is formulated as follows:

$$\min_{\substack{(0 \leq b \leq n-1) \\ (0 \leq p \leq K)}} (E[D(m_b, M_p)])$$

subject to

$$(1) \sum B_{\text{vid}}(m_b, M_p) \leq (B_{\text{max}} - B_{\text{other}}),$$

$$(2) T(m_b, M_p) \leq T_d,$$

where $E[D(\cdot, \cdot)]$ is the expectation of a block distortion. In (1), $B_{\text{vid}}(\cdot, \cdot)$ represents the number of video packets that contribute to the buffer occupancy, whereas B_{max} and B_{other} are, respectively, the maximum buffer size and number of nonvideo packets in the buffer at any given time. On the other hand, $T(\cdot, \cdot)$ and T_d are the time to transmit a video block and total time to render the block on a video player, respectively.

The expectation of the block distortion $E[D(\cdot, \cdot)]$ can be calculated using the mean square error (MSE) between originally transmitted and received blocks:

$$E[D(m_b, M_p)] = \sum_{i=1}^K E[d_i] = \sum_{i=1}^K E\left[\frac{1}{u} \sum_{j=1}^u \{f_{j,i} - \tilde{f}_{j,i}\}^2\right],$$

where $E[d_i]$ is the expectation of the packet distortion, u is the total number of pixels in a packet, and j is the pixel index of the current packet. In (2), f denotes the original image and \tilde{f} represents the reconstructed image.

The IEEE 802.11 medium access control (MAC) standard defines an automatic repeat request (ARQ) mechanism where acknowledgments and timeouts are used to achieve reliable data transmission. If the maximum number of ARQ retries is set to R_{max} , the perceived error-free probability at the application layer is given by

$$\delta_{\text{correct}} = \sum_{r=1}^{R_{\text{max}}} (1 - \delta)\delta^{r-1} = 1 - \delta^{R_{\text{max}}},$$

where δ is the probability of error for transmitted packets over the lossy wireless channel, and r is the index of number of retries. The value of R_{max} is limited for practical purposes, so that the packet transmission delay will not exceed the tolerable delay limit for a video packet. The average number of transmission attempts until a packet is successfully transmitted is given by

$$E[R] = \sum_{r=1}^{R_{\text{max}}} r(1 - \delta)\delta^{r-1}.$$

If round-trip time (RTT) is denoted by T_{RTT} , and packet transmission time is denoted by T_p , the total time to transmit a video block is then calculated by

$$T(m_b, M_p) = N[T_{\text{RTT}}(E[R] - 1) + T_p].$$

2.2. Dynamic programming (DP-) based solution

Because of the NP-hardness of 0-1 knapsack optimization problems [3], suboptimal solutions can be found using DP-based techniques. The time complexity of the DP algorithms

is pseudopolynomial, which is acceptable even for real-time multimedia streaming applications [4]. The computation time can be minimized by limiting the size of a packet and the maximum number of packets in a video block. The first step of DP-based algorithm is to calculate the number of redundant FEC bytes per packet that can minimize the CTR.

Let $H(i)$ denote the number of packets in the playback buffer of the receiving node, right after the playback of the i th video packet, where $i = 1, 2, \dots$. To avoid underflow of the playback buffer, we define a deadline-time (T_c) within which the next video packet should arrive. T_c can be written as

$$T_c(i) = \frac{H(i)}{hf_p},$$

where h is the average number of packets in a video frame, and f_p denotes the video frame rate in frames/second. By extending the work of [1] to the byte-level FEC, the CTR for a packet transmission is defined as

$$\text{CTR}(i) = \frac{qT_c^{-1}(i) + rn}{w(n, k_b)}.$$

Here, q ($0 \leq q \leq 1$) is the cost-time product of transmitting a packet and r ($0 \leq r \leq 1$) is the cost per transmitted byte. In (7), $w(n, k_b)$ is the score of the FEC combination (n, k_b) . Assuming that the channel remains stationary during the transmission of one packet, the score is defined as the total goodput of a selected (n, k_b) strategy. Now, the CTR can be written as

$$\text{CTR}(i) = \frac{qT_c^{-1}(i) + rn}{k_b \text{Pr}(n, k_b)},$$

where $\text{Pr}(n, k_b)$ is the probability that an (n, k_b) code is successful (i.e., at most $n - k_b$ bytes are in error). Let ϵ denote the bit error rate (BER) of the channel and, assuming bit error occurs randomly and independently, the byte error rate (ρ) can be calculated as $\rho = 1 - (1 - \epsilon)^8$. Finally, $\text{Pr}(n, k_b)$ is calculated by

$$\text{Pr}(n, k_b) \triangleq \sum_{i=k_b}^n \binom{n}{i} (1 - \rho)^i \rho^{n-i}.$$

For a given packet size n , and the current channel conditions, a search is made to find the minimum number of FEC bytes required per packet for which the throughput is maximized (i.e., minimize the CTR). The search is stopped when the minimum point of CTR is reached.

The second step of the DP-based algorithm is to calculate p , the optimum number of redundant FEC packets required for each video block with combination M_p . At the packet-level FEC, a fixed RTT is considered in [1], but there is a requirement for a variable RTT because the network loading changes in real-time and channel conditions are not known a priori. As the video packet arrival rate $V(t)$ at the sending node is known and, for a given channel rate $C(t)$, the buffer occupancy $B_{\text{full}}(t)$ of the sending node can be determined as

$$B_{\text{full}}(t) = \int_0^t \{V(\tau) - C(\tau)\} d\tau.$$

Here, we postulate that the number of redundant FEC packets can be calculated as

$$p(t) = \left\lceil K \left\{ 1 - \frac{e^{(B_{full}(t))/\eta}}}{e^{B_{max}/\eta}} \right\} \right\rceil. \quad (11)$$

An exponential function is assumed here such that the number of redundant packets decreases exponentially with time to prevent network congestion due to too many FEC packets, where η is the control parameter. These redundant FEC packets are added to the video block if and only if the second constraint in (1) is met, that is, $T(m_b, M_p) \leq T_d$. Otherwise, the algorithm sequentially takes the values of $p - 1, p - 2, \dots$ until the second constraint is met. This shows that the optimum number of FEC packets is not constant but time-dependent, which is critical for real-time applications.

A packet arriving at the network can be lost due to two independent reasons: first, it is dropped due to limited buffer capacity at the sending node; or second, the packet cannot be recovered by the byte-level FEC combination (i.e., more than $n - k_b$ bytes are in error). The lost packets can be recovered by the packet-level FEC scheme, if the number of lost packets is no more than $N_p - K$, otherwise the video block is said to be lost. Let P_d denote the probability of a packet drop due to limited buffer capacity, the probability of unrecoverable error (i.e., the video block is permanently lost), denoted by P_{ER} , is then given as

$$P_{ER} = \sum_{i=N_p-K+1}^{N_p} \binom{N_p}{i} [\Pr(n, k_b) \cdot (1 - P_d)]^{N_p-i} \cdot [1 - \Pr(n, k_b) \cdot (1 - P_d)]^i, \quad (12)$$

where $\Pr(n, k_b)$ is defined in (9). In (12), i represents the erroneous packets which are not recoverable by packet-level FEC and contribute to the permanent loss of a video block.

3. Numerical Results

Figure 1 shows the number of redundant FEC bytes required for each packet when the channel condition varies from bad to good (i.e., BER varies from 10^{-1} to 10^{-5}). Three packet sizes are considered (i.e., $n = 500, 1000, 2000$ bytes) because the IEEE 802.11 standard allows variable packet sizes with the maximum MAC service data unit (MSDU) size of 2304 bytes. Without loss of generality, we set $q = 1$ and $r = 1$ in (7). The frame rate is set to 30 frames/sec (to emulate the television quality video) and we assume that, on the average, the video frame size is of 5 packets (i.e., $h = 5$), to include the effects of all I-, P-, and B-type video frames. To achieve the desired delay objective for video packets, the maximum number of retries (R_{max}) is set to 4. The number of redundant FEC bytes for which the CTR value is minimized in (8) is determined. By comparing different channel conditions in Figure 1, it is evident that the number of redundant FEC bytes required under bad channel conditions is much higher than that required under good channel conditions, as expected.

To further show the effectiveness of using FEC combinations at both the byte-level and packet-level, consider

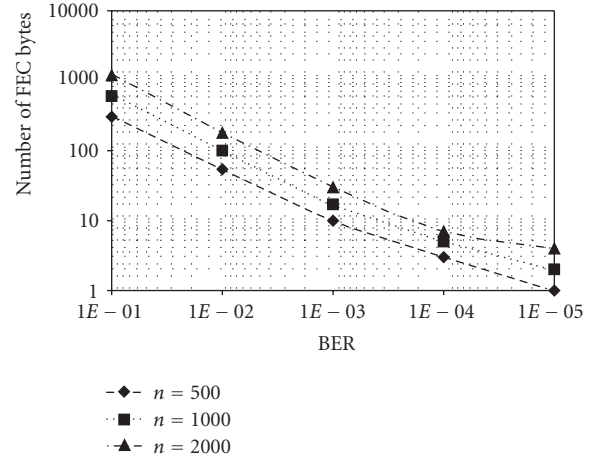


FIGURE 1: Number of FEC bytes required to minimize the CTR.

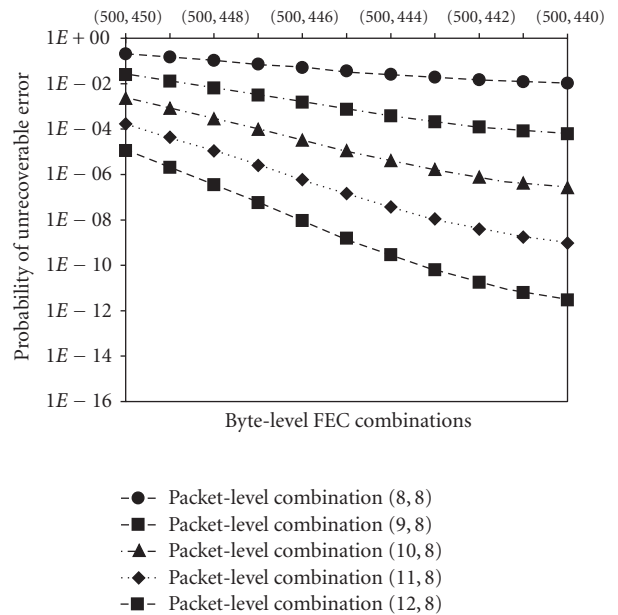


FIGURE 2: P_{ER} for different packet-level FEC combinations.

a scenario where the channel is bad (i.e., $BER = 10^{-2}$), and the packet loss due to sending side's buffer overflow is not very significant (i.e., $P_d = 10^{-3}$). This is a typical scenario where the sending node has enough buffer capacity but unable to transmit packets due to bad channel condition. Figure 2 shows the comparison of byte-level and packet-level FEC combinations (individually and jointly) for reducing the probability of unrecoverable error. Here, we have considered one video block of 8 video packets, that is, $K = 8$, and the packet size is fixed at 500 bytes. As determined from Figure 1, when $BER = 10^{-2}$, the number of redundancy bytes required to minimize the CTR is 54, hence (500, 446) combination is selected at the byte-level. The probability of unrecoverable error can be further reduced from 5.1×10^{-2} at packet-level combination (8, 8) to 3.3×10^{-5} at packet-level combination (10, 8), as shown in Figure 2, by selecting the optimum packet-level

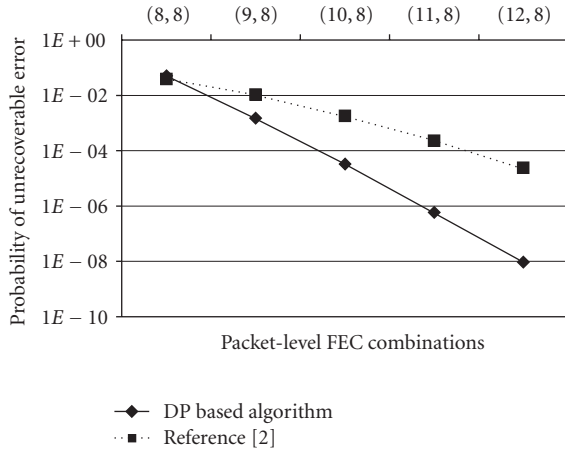


FIGURE 3: Comparison of the P_{ER} of the proposed DP(-based) algorithm and the algorithm presented in [2].

FEC combination allowed by the second constraint in (1). Figure 2 confirms that implementing the FEC at only one level is not good enough for error correction and hence should be supplemented by the FEC implementation at the other level.

The probability of unrecoverable error for the DP-based algorithm proposed in this letter was compared with the corresponding result for the algorithm presented in [2], assuming $n = 500$, $BER = 10^{-2}$, $P_d = 10^{-3}$, and $K = 8$. It is seen from Figure 3 that, under the different packet combinations considered, the probability of unrecoverable error for the DP-based algorithm is lower than that of the algorithm presented in [2]. The CTR model used in the DP-based algorithm introduces a lower bound on the number of FEC bytes, this translates to a better error recovery at the packet-level, thus resulting in lower probability of unrecoverable error.

4. Conclusion

It is concluded that optimum byte-level and packet-level FEC assignment in conjunction with buffer dynamics results in lower probability of unrecoverable error. This translates to high-transmission quality, a desirable requirement for both wireless service subscribers and providers.

Acknowledgments

The authors acknowledge the support of TRILabs, Natural Sciences and Engineering Research Council (NSERC) Canada and the University of Calgary.

References

[1] L. Libman and A. Orda, "Optimal packet-level FEC strategies in connections with large delay-bandwidth products," *IEEE Transactions on Wireless Communications*, vol. 5, no. 7, pp. 1645–1650, 2006.

[2] T.-W. A. Lee, S.-H. G. Chan, Q. Zhang, W.-W. Zhu, and Y.-Q. Zhang, "Optimal allocation of packet-level and byte-level FEC in video multicasting over wired and wireless networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '01)*, vol. 3, pp. 1994–1998, San Antonio, Tex, USA, November 2001.

[3] S. Martello and P. Toth, *Knapsack Problems: Algorithms and Computer Implementations*, Wiley-Interscience Series in Discrete Mathematics and Optimization, John Wiley & Sons, New York, NY, USA, 1990.

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



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

