

MIT Open Access Articles

Packet erasure coding with random access to reduce losses of delay sensitive multislot messages

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

Citation: Zeger, Linda. "Packet Erasure Coding with Random Access to Reduce Losses of Delay Sensitive Multislot Messages." IEEE Military Communications Conference (2009): 1–8. © 2009 IEEE

As Published: <http://dx.doi.org/10.1109/MILCOM.2009.5380121>

Publisher: Institute of Electrical and Electronics Engineers (IEEE)

Persistent URL: <http://hdl.handle.net/1721.1/74261>

Version: Final published version: final published article, as it appeared in a journal, conference proceedings, or other formally published context

Terms of Use: Article is made available in accordance with the publisher's policy and may be subject to US copyright law. Please refer to the publisher's site for terms of use.



PACKET ERASURE CODING WITH RANDOM ACCESS TO REDUCE LOSSES OF DELAY SENSITIVE MULTISLOT MESSAGES

Linda Zeger*
MIT Lincoln Laboratory

August 18, 2009

ABSTRACT

Many commercial and military systems use some form of random access. ALOHA type protocols are particularly useful for multicast traffic and have low complexity; however, they suffer from low capacity and large loss probabilities. The inclusion of packet level erasure coding in single channel ALOHA protocols is a new area. This paper demonstrates a novel combination of a medium access layer tailored to and used with packet level erasure coding; this new protocol not only reduces message loss, but also decreases delays of multislot messages.

It is shown how packet level erasure coding can be used with a tailored random access protocol to render multislot messages more robust against collisions. This resilience to collisions is shown to hold both when limited feedback and no feedback are available. Despite the increased utilization of the collision channel with the additional traffic produced by the erasure code, it is demonstrated to result in a significant decrease in the probability of loss of multislot and single slot messages. The tradeoffs between message loss probability and delays are shown for a range of medium access transmission designs used in conjunction with the erasure coding.

1 INTRODUCTION

Random access protocols are used in many communications systems for sending short data messages or for requesting reservation of a channel. Random access is particularly useful in mobile ad hoc networks, due to the absence of a central base station that could be used to control channel assignments. The high collision probability inherent to random access can result in many lost messages. Best effort delivery would result

in low reliability. Reliable, but not necessarily timely, delivery can potentially be achieved, despite the high collision probability through use of acknowledgments and retransmissions, as well as restriction of the aggregate amount of traffic on the channel.

Investigations to reduce losses and mitigate the impact of contention in random access protocols have been performed. A number of studies [1], [2] have handled errors and erasures in ALOHA channels using Reed-Solomon coding over symbols within a packet. Recently, the idea of coding together symbols from different packets or time slots has been studied in ALOHA systems as well. The idea of using an erasure code at the slot level for multislot messages transmitted with an ALOHA protocol was considered in [3] and [4], for the case of multichannel ALOHA. In these works, multiple fragments of the message can be sent simultaneously on different channels, and capacity is maximized subject to a single delay constraint.

In contrast, we assume only one channel is available for random access in a slotted ALOHA type system, and we examine the probability of successful transmissions as well as the distribution of delays, when a limited number of redundant packets can be transmitted. The limitation in the number of redundant packets controls congestion, reduces the need for feedback, and eliminates messages after times at which they are likely to no longer be useful.

In some random access systems, transmission of acknowledgments may not be practical. One example of the difficulty of the use of feedback occurs with large multicast groups. Hence, we consider the case in which, at most, a limited amount of feedback is used.

In conventional systems with ALOHA type multiple access, an increase in the traffic generated results in an increase in collisions, and hence an increase in packet loss. However, we show that despite the increases in physical layer traffic it produces, packet level erasure correction coding actually *decreases* the probability of upper layer *message* loss.

*This work is sponsored by the United States Air Force under Air Force Contract FA8721-05-C-0002. Opinions, interpretations, recommendations, and conclusions are those of the author and are not necessarily endorsed by the United States Government.

The background and models used for the erasure code and the medium access protocols are specified in Section 2. Simulation results of a slotted ALOHA type model without capture and with limited feedback are presented in Section 3. Analysis and simulation results for a slotted ALOHA type model without feedback and with and without capture are presented in Section 4. Results are summarized in Section 5.

2 MODELS AND BACKGROUND

2.1 ERASURE CODING

We consider the case in which a small number k of original source packets comprise a single message. The length of a transmission time slot is small enough that it accommodates only a single source packet. Hence, a message must be transmitted over several time slots. For example, a message could be an IP packet which is transmitted through a network that uses smaller time slots. The k source packets of the message are coded into n coded packets for transmission. Exactly one coded packet is transmitted in each time slot. Thus with the erasure code, the message now involves transmissions during a total of n time slots, instead of k time slots without the erasure code. We consider coding over the Galois field $GF(2)$, in which case the addition of two packets is the bit-wise XOR of all corresponding pairs of bits in the two packets. The optimal such erasure codes for small k can be derived from [5]. We defer to future work consideration of coding over larger field sizes.

Packet level erasure coding requires at least $k = 2$ source packets. For the case $k = 2$, a single unique parity packet, which is the sum of the two original source packets, can be constructed. We have studied the case of $k = 2$ and $n = 3$, using this parity packet as the third coded packet, where the first two coded packets are the two original source packets. This erasure code is found to decrease the probability of message loss when used with random access. We explored several values of k and n . Generally, for low to moderate offered traffic, the smaller the value of k/n , the greater the decrease in probability of message loss, when random access without feedback as to decoding status is used.

In this paper, the case of $k = 3$ is explored, and since there are exactly 4 distinct parity packets that can be constructed from the 3 original source packets, the value of n can range from 3 to 7. For a given value of n , the optimal code is chosen as the one which minimizes the expected number of the n transmitted packets that need

to be received for successful decoding. The optimal code for $n = 3$ or $n = 4$ enables successful decoding when exactly $k = 3$ of the n transmitted coded packets are received, and $n - k$ are erased. However, for $n = 5, 6$, or 7 there are a number of combinations of $k = 3$ coded packets from the optimal codes that can be received, with the remaining $n - k$ erased, which do not allow successful decoding. In these cases, $k + 1$ packets must be successfully received for decoding. For the case of $n = 7$, the expected number of packets that must be received for successful decoding is 3.2, whereas it is 3.0 for the optimal codes of $n = 3$ or $n = 4$.

When no feedback is used, as in Section 4, then $n = 7$ coded packets are always transmitted for each original 3 packet message. On the other hand, when there is feedback, as in Section 3, only as many of the 7 coded packets as are needed for successful decoding are transmitted. If the original source packets are denoted by a , b , and c , then the order of transmission of the coded packets is a , b , c , $a + b + c$, $a + b$, $a + c$, $b + c$, where $+$ indicates bitwise XOR for corresponding bits in the packets. This order of packets was chosen so as to increase the probability of successful reception with as few of these transmitted coded packets as possible. The order of the first four packets can be interchanged among themselves, as can the order of the last three packets among themselves.

We consider H independently transmitting nodes. Each node transmits the coded packets of a message in the order specified above. The coded packets are not, in general, transmitted in successive time slots, because such bursty transmissions would increase the message loss probability; if one coded packet of a message is lost from collisions, it would be more likely that multiple such packets would be lost if the coded packets are transmitted in bursts. On the other hand, if the packets of the message could be spread out in a stochastic manner, then the probability decreases that enough coded packets of the message suffer collision to prevent decoding.

2.2 MEDIUM ACCESS CONTROL MODEL

If there is no feedback, as in Section 4, all $n = 7$ coded packets are transmitted for each message. For the case with limited feedback discussed in Section 3, initial transmissions of three coded packets, which are stochastically spaced in time, are made. Next, the individual transmissions of zero to four additional coded packets, where the number depends on feedback indicating whether decoding is successful, are made. Af-

ter the third coded packet of a message is transmitted, the transmitting node looks for a 1 bit acknowledgment (ACK) from the receiving node which indicates decoding was successful. If this ACK is received, then the transmitting node ceases transmission of coded packets from this message, and awaits the arrival of a new message to erasure code and transmit. If the ACK is not received, then the fourth coded packet of the current message is transmitted after a random delay. Again, if an ACK is received indicating successful decoding after transmission of the fourth coded packet, transmission of the current message ceases; otherwise, it continues. This process is repeated until either an ACK is received or all $n = 7$ coded packets have been transmitted; in either case, at this point, the transmitting node awaits the arrival of a new message to code and transmit.

After transmission of 7 coded packets, whether or not the message was successfully decoded, transmission of that message is considered complete; if the message was not successfully decoded, then it is considered lost.

The probability that a new message arrives at a given node in a given time slot is denoted by pa . It is assumed that the new message is instantaneously coded and the first coded packet of the new message is transmitted in the time slot of message arrival. In each successive time slot, the next coded packet of the current message that is awaiting access is transmitted with probability pt . A coded packet of the current message awaits access if either there is no feedback and fewer than $n = 7$ coded packets of the message have been transmitted, or there is feedback, and the feedback control protocol detailed above determines that the next coded packet should be transmitted. Once the current coded packet is transmitted, if the next coded packet of the message is awaiting access, then it is transmitted in the next successive time slot with probability pt . Thus, after the first coded packet of a message is transmitted, each subsequent coded packet of that message, when it is at the head of line, is transmitted with probability pt in each time slot; after it is actually sent, the next coded packet advances to the head of line.

If pt is too large, the message loss probability will be higher, as described at the end of Section 2.1 for the case of coded packets transmitted in successive slots, which corresponds to $pt = 1$. Multiple coded packets of a message are more likely to be lost in collisions for larger values of pt than for lower values of pt . If pt is too small, long delays will ensue, and incoming messages can be dropped at the input queue while a current message is completing its transmission. In the simulations, we

thus require

$$1 > pt \gg (n - 1) \times pa \quad (1)$$

for erasure coded messages. Hence, it is unlikely for new messages to arrive at a node before a current message is completely transmitted. On the rare occasions that a new message arrives before the completion of an old message, that new message is dropped from the queue, and it is not counted as a lost message in the results.

2.3 TRANSMISSION OF MULTISLOT MESSAGE WITH BASELINE ALOHA MODEL

In this subsection, use of a conventional slotted ALOHA protocol for transmission, without erasure coding, of multislot messages is described, in order to create a baseline protocol for comparison to erasure coded messages that use the protocol of Section 2.2. The baseline protocol is chosen to minimize the probability of message loss when there is no erasure coding, which results in a classic ALOHA type protocol with $pt = 1$, as described below.

The original k packets of a multi-packet message are to be transmitted without erasure coding for the baseline protocol, so that transmission of the message consists of transmissions in k slots. The entire message is assumed to be lost if any of the k packets are lost.

In contrast to the case in which the messages are erasure coded, when the messages are uncoded, it can be shown that transmission of the k packet message in k successive slots produces a lower probability of message loss than if the k packet transmissions are spaced randomly in time. Thus we consider the baseline protocol to transmit the entire k packet message immediately upon its arrival at a radio.

It is assumed that the first packet of a message is transmitted in the same slot as the arrival, and that each successive packet is transmitted in each successive slot. If two messages overlap by one or more time slots, both messages are assumed lost; this assumption is lifted in Section 4.2. A message will overlap a second message's transmission if its transmission begins during one of the k slots of the second message's transmission or in one of the $k - 1$ slots immediately preceding the second message's transmission. Therefore, the probability that one node's message will overlap that from a second node is $pa \times (2k - 1)$. Thus the probability that none of the $H - 1$ independently transmitting other nodes overlap with a given message from one node is $(1 - pa \times (2k - 1))^{H-1}$. Therefore, the probability of

message loss for the baseline protocol is

$$1 - (1 - pa \times (2k - 1))^{H-1}. \quad (2)$$

Equation (2) is the black dashed curve used in Figures 1 and 5 for comparisons to erasure coded messages transmitted with the random access protocols described in Section 2.2.

3 Performance with Feedback

We first consider the case in which transmission of a multi-packet message is performed with packet level erasure coding and limited feedback, according to the protocols described in Sections 2.1 and 2.2. Since there is no capture here, if two or more nodes transmit coded packets in the same slot, then all the coded packets transmitted in that slot are lost. We consider the case of $H = 20$ independent nodes, and perform simulations of the resulting system.

The probability of loss of an erasure coded message is shown by the green, blue, and red curves in Figure 1. The transmission probability pt in a time slot of the next coded packet within a message is designated by the color of these curves. The black dashed curve represents no erasure coding and the baseline ALOHA model of Section 2.3. It is seen that erasure coding with limited feedback greatly reduces the probability of message loss compared to no erasure coding and the baseline ALOHA transmission protocol, despite the increase in traffic of coded packets and collisions from erasure coding.

The horizontal axis in Figure 1, as well as in the subsequent figures, is the total amount of offered *information traffic* that is transmitted by all H nodes, and is given by:

$$G_i = H \times k \times pa. \quad (3)$$

The actual amount of coded packet traffic transmitted is greater by a factor of n/k . On average, when there is limited feedback, between 3.2 and 4.5 coded packets per message were actually transmitted, depending on both pa and pt . Hence, the actual amount of *coded packets* transmitted can be up to 1.5 times as great as the offered *information traffic* for the limited feedback protocol.

Figure 1 shows that the probability of message loss with erasure coding increases with G_i , quite slowly at first. As discussed in Section 2, increasing pt is seen to increase the probability of message loss. Additional simulations showed that when pt is decreased below .07, the additional decrease in message loss probability is modest over the traffic range shown.

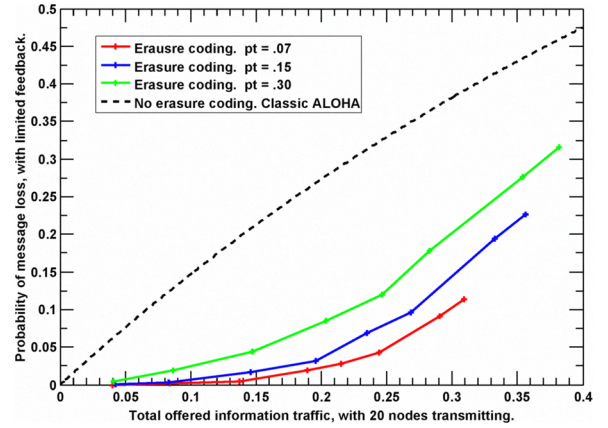


Figure 1: Probability of loss of a 3 packet message, when it is transmitted with an erasure code using between 3 and 7 packets, as determined by the feedback protocol.

However, there is a benefit of using larger pt values, as is shown in Figures 2 through 4, which display simulation results for message delay. Message delay is defined as the total number of slots after the first coded packet of a message is transmitted until the original k packet message is successfully decoded. In the figures described below, the delays are calculated only for those messages that are successfully decoded; lost messages are not included in the delay calculations. Delays from processing, propagation, and feedback mechanisms are neglected, so that message delay is due only to the medium access delay for transmission of every coded packet. Thus the greater the total number of coded packets that must be transmitted for successful message decoding, the greater the delay will be. When there are more collisions of coded packets, which occurs at higher G_i values, more coded packets must be transmitted for successful message decoding.

The mean message delay is plotted in Figure 2 for the same three values of pt as used in Figure 1. It is seen that while the lower the value of pt , the lower the probability of message loss, a lower value of pt also entails a significantly higher mean message delay, due to the longer transmission times between successive coded packets of a message. For a fixed pt , the mean message delay increases with traffic loading, due to an increase in collisions, and therefore, an increase in the number of coded packets that must be transmitted.

The probability of a message satisfying a delay constraint is obtained from the cumulative distribution function (CDF) of the message delay. For example, we consider delay constraints of 15 and 60 time slots.

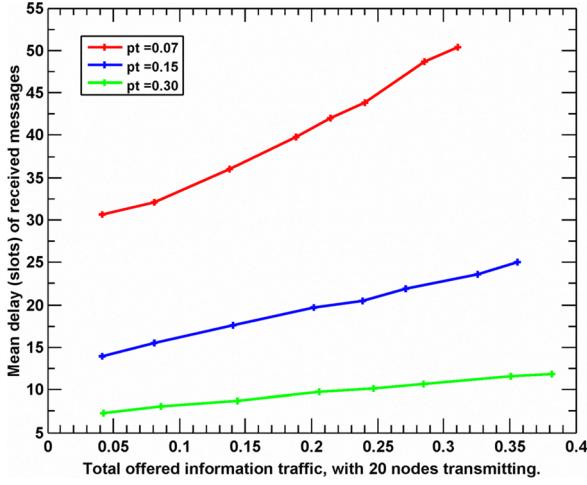


Figure 2: Average delay of erasure coded received messages.

The CDF at these two values is plotted vs. G_i in Figures 3 and 4 respectively. These figures show that the probability of satisfying a specified delay requirement decreases with increasing G_i for fixed pt ; a greater G_i results in greater numbers of collisions, which then require transmission of more coded packets, thereby causing greater delays. These figures also show that for fixed G_i , the probability of satisfying a delay requirement decreases as pt decreases; as pt decreases, the random times between transmission of successive coded packets of a message increase. Figure 4 shows that over the range of G_i values displayed, for $pt = .15$ or $.3$, more than 97% of all messages meet the 60 slot delay requirement, whereas for $pt = .07$ between 66% and 90% of the messages meet this delay requirement. Figure 3 illustrates the comparatively lower probabilities of satisfaction of a 15 slot message delay requirement over the same range of G_i : for $pt = .3$ between 68% and 91% of the messages satisfy this more stringent delay requirement, while for $pt = .07$ only between 9% and 26% of the messages satisfy it.

Although multi-packet messages were considered here, it should be noted that the probability of message loss for single packet messages, which is the probability that a single packet experiences a collision, is approximately given by the value of the horizontal axis of Figure 1. Therefore, it is seen that transmission of individual single packet messages results in a higher message loss probability than if they are grouped together and erasure coded.

We have shown that when there is no capture, but there is limited feedback, erasure coding results in a significant reduction in probability of message loss. This

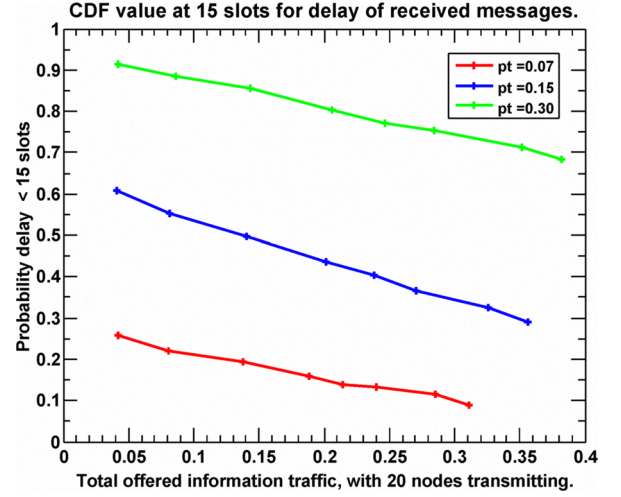


Figure 3: CDF value at a 15 slot delay vs. G_i .

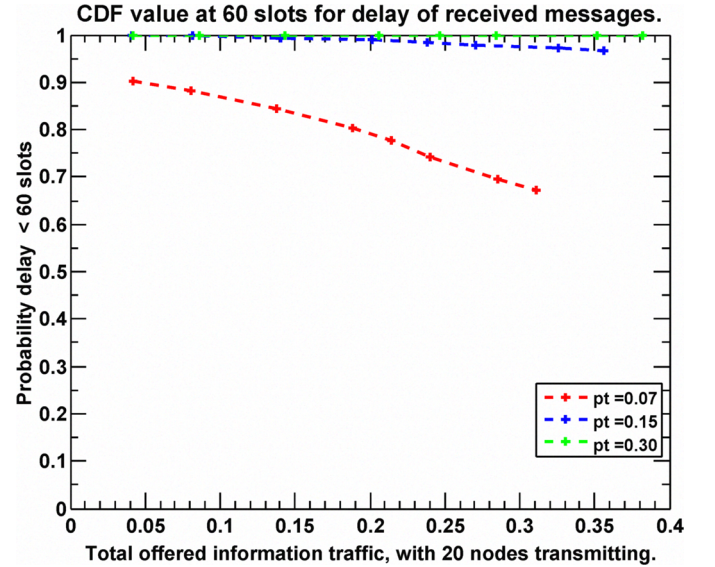


Figure 4: CDF value at a 60 slot delay vs. G_i .

decrease in message loss probability can be slightly compromised, so as to satisfy specified delay requirements by adjusting pt to achieve the desired tradeoff between loss probability and delay.

4 PERFORMANCE WITHOUT FEEDBACK

In a number of situations, for example with large multicast groups or large round trip times, feedback may not be desirable, or even feasible. In these cases, the value of n , the total number of coded packets transmitted for each message, must be fixed *a priori*, and n coded packets are always sent, regardless of whether the message can be decoded after transmission of fewer than n coded packets. We again specifically consider the case of $k = 3$, but now $n = 7$ coded packets are transmitted for *every* message. For much of the range of the G_i values considered here, the probability of message loss is lower when $n = 7$ packets are sent than for uses of smaller values of a fixed n . However, for larger values of G_i , when the utilization and contention are already high, smaller values than 7 for the fixed n can give better performance. In the remainder of this section, we display results for $n = 7$. The medium access model described in Section 2.2 is again used here for erasure coded messages.

4.1 NO CAPTURE

In this subsection, as in Section 3, the assumption is that if there is more than one coded packet transmitted in a slot, all packets transmitted in that slot are lost.

Unlike the case in which there is feedback as in Section 3, here and in Section 4.2 there will be many times in which more coded packets are sent than are needed by a single receiver. Hence for the same value of G_i , there will be more offered *coded* traffic without feedback, and thus many more collisions of coded packets, than when feedback is available. Thus performance for the results shown in Figure 5 is moderately degraded, relative to the case of limited feedback discussed in Section 3.

However, the message loss probability is still significantly lower than for transmission without erasure coding using the baseline slotted ALOHA protocol of Section 2.3, as represented by the black dashed curve. Thus we have demonstrated a gain in use of erasure coding with random access, even when no feedback is available.

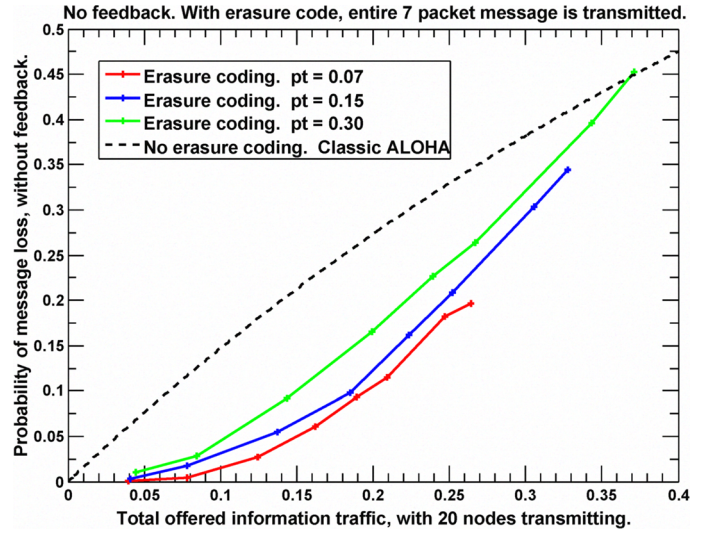


Figure 5: Probability of loss of a 3 slot message transmitted with an erasure code using 7 slots.

4.2 CAPTURE

Finally, we consider the case of capture, in which even with collisions in a slot, the packet may not be lost. The capture effect occurs when two or more signals overlap at a receiving node and that node is able to successfully receive the signal of greater power. The capture probability, denoted by p_{cap} , is the probability that a single transmitted packet is received by a node, given that at least one other packet is transmitted by another node in the same time slot.

In order to obtain an approximation of the performance with capture, we first construct a baseline protocol, by again considering a slotted ALOHA type model and extending the baseline model of Section 2.3 to include capture. Hence, the baseline protocol here also has no erasure coding, and the k packets are transmitted in successive slots. Thus the overlap of two or more messages by one or more time slots results in a collision. Due to the capture effect, the collision here, unlike that in Section 4.1, does not necessarily result in packet loss, and hence message loss, for all messages in the collision.

In this section for both the baseline protocol and the erasure coded model, it is assumed that the aggregate arrival process of *messages* from all transmitting nodes can be approximated by a Poisson process. It is further assumed that the signal strength is relatively constant over k successive slots. Thus in our calculation, if a collision occurs on a packet in one slot of the message and that packet is successfully received through the capture effect, then all other slots of the message are assumed to be also successfully received, even though they may ex-

perience collisions as well. Since this assumption does not necessarily hold if a collision involves more than two messages, it slightly undercounts the number of message losses. Therefore, our probability of message loss shown below for the baseline protocol is actually a *lower bound* for this protocol.

Thus it can be shown that a lower bound for the probability of loss of a k packet message with this baseline protocol and the capture effect is

$$P_{base} = (1 - p_{cap}) \times [1 - \exp(-(2 - 1/k) \times G_i)] \quad (4)$$

Equation (4) is an extension of (2) to include the capture effect, and also to model the aggregate arrival process of the messages (each consisting of k packets) from all nodes as a Poisson process. As the number of nodes H in (2) becomes large and as p_{cap} approaches zero, equation (2) approaches equation (4).

Next we construct a model for the erasure coded messages, which have n coded packets transmitted without feedback for each original k packet message. We assume the transmissions of the n coded packets are randomly spaced in time, as in Section 2.2. Furthermore, it is also assumed that pt is small enough that the aggregate transmission process of coded packets from all nodes can be approximately modeled as a Poisson. In this case, the probability of losing a coded packet of a message is

$$p_{packet} = (1 - p_{cap}) \times [1 - \exp(-(n/k) \times G_i)]. \quad (5)$$

Therefore, the probability that the message is not successfully decoded is

$$P_{message} = 1 - \sum_{j=k}^n \binom{n}{j} p_{packet}^{n-j} \times (1 - p_{packet})^j \times h(j) \quad (6)$$

where $h(j)$ is the probability that the k packet message can be decoded if a random selection of j of the n coded packets are received. For $n = 7$ and $k = 3$, receipt of three coded packets will be sufficient for message decoding for 80% of the possible combinations, while four coded packets will be needed the remainder of the time; hence, in this case $h(j) = 1$ for $j > 3$, while $h(3) = .8$.

Equations (4), (5), and (6), are used to construct the probability of message loss with capture when erasure coding and random packet spacing are used, as well as when the baseline protocol is used. Figure 6 displays the results for $k = 3$, $n = 7$, and $p_{cap} = .4$. The probability of message loss when only the $k = 3$ original packets are transmitted without erasure coding, according

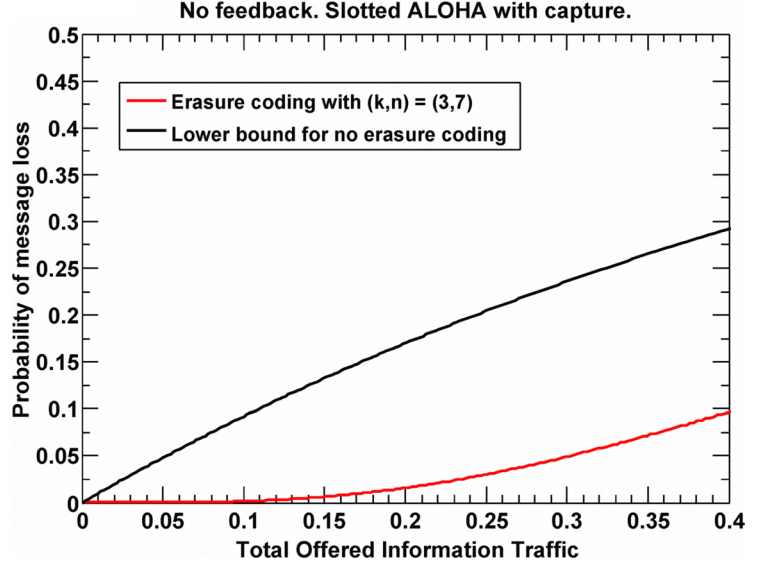


Figure 6: Probability of loss of a 3 packet message with (red) and without (black) erasure coding to 7 packets, when there is capture.

to the baseline protocol, is shown by the black curve. The probability of message loss when each message is transmitted with the fixed $n = 7$ coded packets is shown by the red curve. The capture effect improves the baseline uncoded (black curve) performance as compared with baseline uncoded (black) curves of Figures 1 and 5. The use of the erasure code with tailored random access further greatly decreases the probability of message loss, so that low loss levels are obtained for moderate values of G_i .

The results shown thus far have considered a single receiver. For multicast, different users may receive different packets depending on their relative locations. Thus transmitting a large enough fixed number of coded packets without feedback can be even more beneficial in the multicast scenario. The potentially greater gains that may be achieved with multicast are left to future work.

5 SUMMARY

Use of packet level erasure coding greatly reduces probability of loss of multislot messages when employed with a random access protocol that is tailored to the erasure coding. The tradeoff between delays and probability of message loss was shown to be determined by the precise medium access parameters. With judicious choice of these parameters, delays can be mitigated. The effectiveness of erasure coding was shown to be large for systems with limited feedback. Erasure cod-

ing was still significantly effective when there is no feedback, both when the receiver does and does not exhibit the capture effect.

References

- [1] M. B. Pursley, "Delay and Throughput for Three Transmission Schemes in Packet Radio Networks", IEEE Trans. Comm., Vol. 37, No. 12, Dec. 1989, p. 1264-1274.
- [2] S. Fakhouri and A. Yongacoglu, "Coding and Multiple Access Techniques for Data Transmission in Land-Mobile Satellite Networks", IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing, May 1991, p. 410-413.
- [3] D. Baron and Y. Birk, "Coding Schemes for Multislot Messages in Multichannel ALOHA with Deadlines", IEEE Trans. on Wireless Comm., Vol. 1, no. 2, April 2002, p. 292-301.
- [4] Y. Birk and Y. Keren, "Judicious Use of Redundant Transmissions in Multichannel ALOHA Networks with Deadlines", IEEE JSAC, vol. 17, no. 2, Feb. 1999, p. 257-269.
- [5] J. Plank, A. Buchsbaum, R. Collins, M. Thomason, "Small Parity-Check Erasure Codes - Exploration and Observations", Technical Report UT-CS-04-537, University of Tennessee Dept. of Computer Science, November 2004. J. Plank, "Enumeration of Small, Optimal and Near-Optimal Parity-Check Erasure Codes", Technical Report UT-CS-04-535, University of Tennessee Dept. of Computer Science, November 2004.