Priority Encoding Transmission

Andres Albanese* Johannes Blömer† Jeff Edmonds‡ Michael Luby§
Madhu Sudan¶

Abstract

We introduce a novel approach for sending messages over lossy packet-based networks. The new method, called Priority Encoding Transmission, allows a user to specify a different priority on each segment of the message. Based on the priorities, the sender uses the system to encode the segments into packets for transmission. The system ensures recovery of the segments in order of their priority. The priority of a segment determines the minimum number of packets sufficient to recover the segment.

We define a measure for a set of priorities, called the *rate*, which dictates how much information about the message must be contained in each bit of the encoding. We develop systems for implementing any set of priorities with rate equal to one. We also give an information-theoretic proof that there is no system that implements a set of priorities with rate greater than one.

This work has applications to multi-media and high speed networks applications, especially in those with bursty sources and multiple receivers with heterogeneous capabilities.

*International Computer Science Institute, Berkeley, California.

†International Computer Science Institute. Berkeley, California. Partially supported by NSF operating grant CCR-9304722, and ESPRIT BR Grant EC-US 030.

‡International Computer Science Institute, Berkeley, California. Supported by an NSF postdoctoral fellowship and by a Canadian NSERC postdoctoral fellowship.

§International Computer Science Institute, Berkeley, California. Partially supported by NSF operating grant CCR-9304722, Israeli-U.S. NSF BiNational Science Foundation grant No. 92-00226 and ESPRIT BR Grant EC-US 030.

¶IBM Thomas J. Watson Research Center, Yorktown Heights, New York. Participated in research while visiting the International Computer Science Institute.

1 Introduction

In many multi-media applications, long messages are to be transmitted in real-time across multiple network links. A message is not sent as one unit, but broken into small packets that are sent through the medium. Bit corruption may occur in packets due to transmission, but these can be handled on a link-by-link basis using error correcting techniques. Thus, we can assume that packets are indivisible units that arrive intact if they arrive at all. Once the packets are sent, some of the packets may arrive promptly, but arbitrary subsets of packets may be lost or delayed beyond the point of usefulness due to global conditions in the network such as congestion, buffer overflows and other causes. We hereafter call media with this property *lossy* media. At some point in time, the receiver cannot wait for packets any longer and must recover as much of the original message as possible from the packets received.

It seems highly plausible that packet loss as described will be an ordinary phenomena for reasonably priced networks that connect millions of users spread around the world simultaneously running a multitude of high bandwidth real-time applications. Furthermore, packet losses will not be spread uniformly over the network, but may vary between different sites and may fluctuate over time. Thus, it could be argued that, analogous to noise being the nemesis of analog communication, and error being the nemesis of digital communication, loss will be the nemesis of packet-based wide-area real-time communication.

This paper proposes a general and flexible method to cope with packet loss, which we call Priority Encoding Transmission (PET). The user partitions the message into segments and assigns each segment a priority. Based on their priority, the segments are encoded into a set of packets. The priority of a segment specifies the minimum number of packets sufficient to decode it. The system guarantees that a segment can be decoded from any subset of packets as long as the number of packets in the subset is at least equal to the segment priority.

In the networking community encoding systems

which allow recovery of the message from only a subset of packets of the encoding have been proposed, for example a system based on Reed-Solomon-code was suggested by [8, McAuley] and empirically evaluated by [4, Biersack]. A similar encoding system has been proposed by [9, Rabin]. He uses essentially the same coding techniques that are used in this paper. However, these systems allow only one priority level for the entire message.

[10, Shacham] also suggests methods for sending prioritized messages over networks. However, those methods require computation of channel capacities from the sender to each receiver, which may be impractical for very large networks with capacities that vary quickly because of congestion. Furthermore, this work does not handle packet losses.

Section 2 describes potential applications of the PET system to transmit multicast video images over heterogeneous networks, to encode IP packets for the recovery of the header and control information from a partial delivery of ATM cells, and to increase the quality of service (only the packet loss) provided by the network layer to an application.

Section 3 gives a formal definition of PET systems and it describes properties for deterministic and probabilistic models. A deterministic PET system is described in Section 4, and a probabilistic system is described Section 5. Section 6 defines a geometric measure of information and gives an information-theoretic proof that the rate of any PET system is at most 1.

2 Applications

Priority Encoding Transmission is a new method for sending information messages through a lossy transmission system to multiple receivers. Depending on processing power, each receiver decodes the most important information from partially received messages.

Present networks use multiple window protocols to retransmit missing information for communicating with multiple receivers. Consequently, the information rate is determined by the worst case receiver. Future information highways will provide an ever wider range of performance due to the proliferation of wide area networks and broadband technology. Information from a sender must be received by all users participating in the multicast session. Furthermore, each user should be free to select among the available transport services and receiving stations.

Senders specify how to assign priority levels to information objects, applications arrange the objects inside information blocks, and the different objects within a block are encoded to produce the multiple packets to be transmitted over the unreliable media. Depending on the number (or percentage) of received segments within the block, a number of objects are decoded by their priority level.

PET systems techniques can be used in several of the network layers of the protocol architecture. This section describes two possible applications: multicasting of video images over heterogeneous networks, and assembly layers in ATM networks.

2.1 Video multicasting over heterogeneous networks

High quality images may consist of as much as 96 Mbits per image, and images may be sent at the rate of 30 per second. If some of the packets containing the image are delayed or lost, the receiver cannot delay displaying the video image. An unfavorable scheme would be to physically partition the image into small regions and place in each packet the information about a single region. The resulting displayed image could be displeasing, consisting of a patchwork of high resolution regions corresponding to received packets intermixed with blank regions corresponding to lost packets.

Using JPEG or MPEG, a discrete cosine transform can be applied to a video image to produce what is hereafter called a message [11, Wallace], [6, Le Gall]. Besides allowing a highly compressed representation of the image, this message has a nice property. Consider ordering the information in the message so that the lowest frequency coefficients come first followed by successively higher frequency coefficients. The nice property is that the quality of the image that can be reconstructed from a prefix of this ordered message improves gracefully as a function of the length of the prefix. A PET system can be used to send this real-time prioritized information over a media that sometimes loses and/or inordinately delays delivery of packets.

2.2 Assembly layers in ATM networks

Broadband communication systems using ATM (Asynchronous Transfer Mode) techniques can carry IP (Internetwork Protocol) packets. We propose the encoding of multimedia applications using IP packets to guarantee the delivery of highest-priority data or the timely recovery of real-time data when IP packets are lost. Voice, data, and video can be coded into the same message with different priorities to guarantee a quality of service (packet loss only) required by each media in the message. Another possible application is

in the ATM adaptation layer to encode IP packets into ATM cells in such a way that the IP packet headers and other control information in the packet are recovered with higher priority when ATM cells are lost. In both cases, PET can be used to recover either missing packets or cells.

3 Definitions of PET systems

We assume information is transmitted over a medium in units of packets of bits. The medium is *lossy*, i.e., transmitted packets may get lost. It is assumed that either a packet is completely received or is completely lost. There is no assumption made about which packets are received or lost, i.e., no guarantee is given that certain packets make it to the receiver or do not make it to the receiver. The packets may also arrive in any order.

Convention: *Throughout this paper we assume that each packet has a unique identifier, that distinguishes it from the remaining packets. The number of bits necessary to write down this identifier is not included in the packet size.*

This convention is justified by the fact that including a unique identifier into a packet does not require a lot of bandwidth.

If a message M is to be transmitted over a lossy medium the goal is to encode the message M into a code $E(M)$ which is then sent to the receiver. The encoding is such that the receiver is able to recover parts of the original message without receiving the entire encoded message. Moreover, the sender should be able to assign different priorities to different pieces of the message and the receiver should be able to recover the pieces of the message in order.

3.1 Definition of a deterministic PET system

Let M be a message of length m to be sent over a lossy network.

Definition 3.1 (PET system) *A PET system with message length m, packet size ℓ, n packets, and encoding length $e = n\ell$ consists of the following:*

(i) An encoding function *E that maps a message M of length m onto an encoding $E(M)$ consisting of n packets, i.e. $n\ell$ bits.*

(ii) A decoding function *D that maps a set of at most n packets onto a bit string of length m.*

(iii) A priority function *β that maps $[1..m]$ to integral multiples of ℓ.*

The guarantee of the system is that, for all messages M of length m and for all $i \in [1..m]$ D decodes the i^{th} bit of the message from any subset of packets of the encoding $E(M)$ that contain in total at least β_i bits.

Throughout this paper we assume without loss of generality that the priority function is monotonically increasing, i.e., $\beta_1 \leq \beta_2 \leq \cdots \leq \beta_m$. Thus, β_i can also be thought of as the number of bits needed to recover the first i bits of the message. The values of β are given in terms of multiples of ℓ, since it is assumed that only complete packets of bits are acquired.

An important information-theoretical measure for a PET system is how much information each bit in the encoding contains about the message.

Definition 3.2 (Rate of a priority function and a PET system) *For a function β mapping $[1..m]$ to the natural numbers, the* rate *of β is*

$$\text{rate}_\beta = \sum_{i \in [1..m]} 1/\beta_i.$$

The rate of a PET system is the rate of its priority function.

Intuitively, in a PET system with priority function β, each β_i bits of the encoding must determine the i-th message bit M_i. Hence on average each encoding bit contains $1/\beta_i$ bits "about" M_i. Therefore, on average each bit of the encoding contains $\text{rate}_\beta = \sum_{i \in [1..m]} 1/\beta_i$ bits in total "about" the message. However, a single bit can contain at most one bit of information. Hence, it is reasonable to expect that such a system is possible only if $\text{rate}_\beta \leq 1$, and we prove this to be the case in Section 6 (Theorem 6.5). On the other hand, we show in Section 4 (Theorem 4.3) that, for a given priority function β with $\text{rate}_\beta = 1$, a PET system with priority function γ can be constructed such that $\text{rate}_\gamma = 1$ and such that, for all $i \in [1..m]$,

$$\gamma_i \leq (1 + 5/\alpha) \cdot \beta_i.$$

Here, $\alpha \geq 3$ is an adjustable parameter that balances the tradeoff between the closeness of the approximation of γ to β, the total encoding length, and the packet size.

3.2 Definition of a probabilistic PET system

We only highlight the differences between probabilistic and deterministic PET systems. The main differences are that there is a random string R shared by both the sender and receiver that is used to encode and decode messages and a failure probability $p > 0$. The string $R \in \{0,1\}^r$ is used to select an encoding/decoding pair (E^R, D^R) from a family of 2^r such pairs. Once this pair has been selected the encoding and decoding is deterministic. The guarantee of a probabilistic PET system with priority function β is that, for all messages M of length m, for all $i \in [1..m]$, and for any subset of packets that contain in total at least β_i bits, if the function E^R was used for the encoding then with probability at least $1-p$ the function D^R decodes the i^{th} bit of the message from this subset. The probability is with respect to the uniform distribution on the random string $R \in \{0,1\}^r$. We stress that this probability is not over a particular distribution over the messages. For any fixed value of R, an encoding/decoding pair succeeds or fails on certain subsets of packets, independent of the message.

In Section 5 we describe a procedure that, given a priority function β with rate one returns a PET system that satisfies a family of failure probability/priority function pairs. These pairs are parameterized by $\delta > 0$ and for each δ the priority function is $(1+\delta)\beta$ and the failure probability is of the form $\exp(-\delta^2 t/16)$ for some parameter t. Hence the failure probability drops exponentially with increasing δ. For each δ the priority function has rate $1/(1+\delta)$ and hence the parameter δ establishes a tradeoff between the rate and the failure probability. The parameter t is used to obtain a tradeoff between the confidence and the efficiency of the PET system. For different values of δ the PET system only differs in the number of packets sent, i.e, in the length of the encoding.

4 A PET system

We describe a general method that takes any given priority function β and produces a PET system which has a priority function that closely approximates β. The method works by first partitioning the message into blocks based on the priority function β, and then using the partition to implement a PET system based on erasure codes.

In the first subsection we describe erasure codes. In the second subsection, we assume we have the partitioned message and show how to implement a PET system based on erasure codes. Finally, we describe an algorithm that accepts the description of an arbitrary priority function β and produces a partitioned message. The PET system that results from combining these parts has a priority function which closely approximates β.

4.1 Erasure codes: A basic encoding system

An erasure code is specified by a triple $\langle b, n, w\rangle$, where $n \geq b$. It encodes a message M of length $m = wb$ into a code E of length $e = wn$. Both the message and the code consists of words of length w each. The code has the property that all b words of M can be recovered from any b words of E.

Descriptions of erasure codes can be found, for example, in [9, Rabin]. One implementation of erasure codes is the following. The b words of M are viewed as the coefficients of univariate polynomial of degree $b-1$ over GF[2^w]. Call this polynomial G. The j^{th} word of the code consists of the value of the polynomial G evaluated at the field element $j \in$ GF[2^w]. Since G is of degree $b-1$, any b words (together with the indices of the words) uniquely determine G. The message M, i.e., the coefficients of G, can be recovered from any b words by interpolation.

This implementation requires that $n \leq 2^w$, or equivalently that

$$w \geq \log(n). \tag{1}$$

This ensures that there are at least n different elements in the field GF[2^w] on which to evaluate the polynomial.

4.2 Block systems

The first step in constructing a PET system given a priority function β is to partition the message into blocks based on β. This first step is described in the next subsection. In this subsection, we show how to implement a PET system given a partition of the message. In this and all subsequent constructions, we ignore small roundoff errors.

Definition 4.1 (m-partition) *An m-*partition *consists of a sequence of positive integers*

$$\langle m_1, \ldots, m_d\rangle$$

such that

$$\sum_{j\in[1..d]} m_j = m.$$

Lemma 4.2 *Given an m-partition $\langle m_1, \ldots, m_d \rangle$, a PET system with priority function γ can be constructed with the following properties:*

(i) *For all $j \in [1..d]$, $\gamma_j = dm_j$.*

(ii) $\text{rate}_\gamma = 1$.

(iii) *The encoding length is $e = \max_{j \in [1..d]}\{\gamma_j\}$.*

(iv) *The packet size is $\ell = dw$, where $w = \log(e)$ is the word size.*

Proof of Lemma 4.2: Let $B_1, \ldots, B_d$ be the blocks of M, and thus the length of B_j is m_j. The basic idea is to use a separate erasure code for each of the d blocks of the message. The j^{th} erasure code is used to encode B_j into a code E_j consisting of n words, each of length w, where n and w are fixed below. Thus, B_j consists of $b_j = m_j/w$ words. The entire encoding E consists of n packets of size $\ell = wd$ each, where the k^{th} packet consists of the concatenation, for $j \in [1..d]$, of the k^{th} word from the code E_j. Thus, the code length is $e = \ell n$. The decoding works in the obvious way.

Since we use an erasure code for each block, all bits in the same block have the same priority. Any b_j words of the code E_j suffice to recover block B_j. Since there is one such word in each packet, it follows that b_j packets of E are sufficient to recover B_j. Thus, the priority of all bits in block B_j is

$$\gamma_j = \ell b_j = dm_j. \tag{2}$$

This proves item (i). Note that

$$\text{rate}_\gamma = \sum_{j \in [1..d]} m_j/\gamma_j = 1. \tag{3}$$

This proves item (ii). To ensure that the entire message can be recovered from all the packets, we need

$$n \geq \max_{j \in [1..d]}\{b_j\}. \tag{4}$$

With the number of packets set to make this an equality, the total encoding length is $e = \ell n = \max_{j \in [1..d]}\{\gamma_j\}$. This proves item (iii). To use the implementation of erasure codes described in Section 4.1, we need the word length w to be at least $\log(n)$ from Inequality (1). Thus, we can set $w = \log(e) \geq \log(n)$. This proves item (iv). □

In the system described above, each packet needs to contain an identifier which is interpreted as the field element value at which the d message blocks considered as polynomials are evaluated. Although this is part of the packet, we did not include it in the packet size. The overhead per packet because of this is at most w bits.

4.3 Partitioning a message

In this subsection, we show how to construct an m-partition based on a message length m, a priority function β with $\text{rate}_\beta = 1$, and a parameter $\alpha \geq 3$. When this m-partition is used to construct a PET system as described in Lemma 4.2, the priority function γ of the system is a close approximation of β. The parameter α is used to balance the tradeoff between the closeness of the approximation of γ to β, the total encoding length, and the packet size.

We first give the main theorem (Theorem 4.3) and then give the partitioning lemma (Lemma 4.4) upon which the theorem is based.

Theorem 4.3 *Let β be a priority function with $\text{rate}_\beta = 1$ for messages of length m. There is an efficient algorithm that, on input β, m, and a value $\alpha \geq 3$, produces a PET system with priority function γ with the following properties:*

(i) *The encoding length e is at most $3\alpha m$.*

(ii) *The packet size ℓ is at most $\alpha^2 \log^2(3\alpha m)$.*

(iii) *For all $i \in [1..m]$, $\gamma_i \leq (1 + 5/\alpha) \cdot \beta_i$.*

(iv) $\text{rate}_\gamma = 1$.

Proof of Theorem 4.3: The first step is to partition M based on β, m and α as described in Lemma 4.4. We then use Lemma 4.2 to get the PET system. It is easy to verify that it has claimed properties. □

Note that as α increases the closeness of the approximation of γ to β improves whereas the encoding length and the packet size both increase. Little attempt is made in the theorem to optimize the minimal value $\alpha = 3$ for which the result holds or the other absolute constants associated with α.

In practice, the transmission medium dictates the size ℓ_{medium} of a packet. If $\ell < \ell_{\text{medium}}$ then the packet size of the PET system can be easily scaled up to ℓ_{medium}.

Lemma 4.4 *Let β be a priority function with $\text{rate}_\beta = 1$ for messages of length m. There is an efficient algorithm that, on input β, m, and a value $\alpha \geq 3$, produces an m-partition $\langle m_1, \ldots, m_d \rangle$ that satisfies the following properties:*

(i) $\max_{j \in [1..d]}\{m_j\} \leq 3\alpha m/d$.

(ii) For all indices i in the j^{th} block B_j, $m_j \leq (1 + 5/\alpha) \cdot \beta_i/d$.

(iii) $d = \alpha^2 \log(2\alpha m)$.

Proof of Lemma 4.4: To satisfy part (i) of the lemma, we first introduce an intermediate priority function β'. For all $i \in [1..m]$, let $\beta'_i = c' \cdot \min\{\beta_i, \alpha m\}$, where $1 \leq c' \leq 1+1/\alpha$ is a small normalizing constant that makes $\text{rate}_{\beta'} = 1$. Note that for all i,

$$\beta'_i \leq c' \beta_i. \tag{5}$$

Furthermore,

$$\beta'_m \leq c' \alpha m \leq 2\alpha m. \tag{6}$$

We set $d = \alpha^2 \log(2\alpha m)$, which satisfies part (iii) of the lemma. We also set two intermediate parameters $c = 1+1/\alpha$ and $k' = \alpha \log(2\alpha m)$ and define $k = d - k'$. These parameters are set so as to satisfy the following:

$$c^{k'} \geq 2\alpha m. \tag{7}$$

$$k = d(1 - 1/\alpha). \tag{8}$$

Inequality (7) holds for any $\alpha \geq 2$. Based on these settings of parameters, we then iteratively cut the message into blocks $B_1, \ldots, B_d$ as follows, where i_j denotes the first index in block B_j and m_j is the length of B_j. Suppose that indices $i_1, \ldots, i_j$ have already been set. Then i_{j+1} is set to be the smallest index greater than i_j that satisfies at least one of the following two conditions:

Condition 1: $\beta'_{i_{j+1}} > c\beta'_{i_j}$.

Condition 2: $\sum_{i=i_j}^{i_{j+1}} 1/\beta'_i > 1/k$.

We first verify that the entire message is completely partitioned into the d blocks. From $\beta'_1 \geq 1$, from Inequality (6), and from Inequality (7), Condition (1) can happen at most k' times. Because $\text{rate}_{\beta'} = 1$, Condition (2) can happen at most k times. Thus, the total number of blocks used to partition the entire message is at most $k + k' = d$.

We now derive an upper bound on the number m_j of bits in block B_j. By Condition (1), for all $i \in B_j$, $\beta'_i \leq c\beta'_{i_j}$. By considering the worst case, i.e., when this is equality for all $i \in B_j$, and using Condition (2), it follows that $m_j \leq c\beta'_{i_j}/k$. From this and from Inequality (5), it follows that, for all indices i in block B_j,

$$\begin{aligned} m_j &\leq c \cdot (d/k) \cdot \beta'_{i_j}/d \leq c \cdot (d/k) \cdot \beta'_i/d \\ &\leq c' \cdot c \cdot (d/k) \cdot \beta_i/d. \end{aligned} \tag{9}$$

Note that $c' \leq 1 + 1/\alpha$, $c = 1 + 1/\alpha$, and $d/k \leq 1/(1-1/\alpha)$, It can be easily shown that, for all $\alpha \geq 3$,

$$(1 + 1/\alpha)^2/(1 - 1/\alpha) \leq 1 + 5/\alpha. \tag{10}$$

From this inequality, and from Inequality (9), it can be seen that for all indices i in B_j, $m_j \leq (1+5/\alpha) \cdot \beta_i/d$. This satisfies part (ii) of the lemma.

Because $m_j \leq c \cdot (d/k) \cdot \beta'_{i_j}/d$ from Inequality (9), and because $\beta'_{i_j} \leq \beta'_m \leq c'\alpha m$ from Inequality (6), it follows that

$$\max_{j \in [1..d]} \{m_j\} \leq c' \cdot c \cdot (d/k) \cdot \alpha m/d.$$

Thus, from Inequality (10), and because $\alpha \geq 3$ implies that $1 + 5/\alpha \leq 3$, it follows that $\max_{j \in [1..d]}\{m_j\} \leq 3\alpha m$, proving part (i) of the lemma. □

5 A probabilistic PET system

Erasure codes, as described in Section 4.1 and used in the deterministic PET system in Section 4.2, are specified by a triple $\langle b, n, w \rangle$. Recovering the message M of length $m = wb$ requires the interpolation of polynomial of degree $b-1$ over $\text{GF}[2^w]$, and for large values of b and w this may turn out to be impractical. In this section we briefly describe a probabilistic PET system that is based on probabilistic erasure codes. These codes allow a smaller word size and smaller degree polynomials. The idea is to break the message into fixed size pieces, called *bundles*, of $t < b$ words each. The encoding is probabilistic in the sense that given any $(1+\delta)b$ words of the code a bundle of the message can be decoded with some probability depending on δ. However, the decoding of a bundle involves only the interpolation of a degree $t-1$ polynomial over $\text{GF}[2^w]$.

A straightforward method to do this is to choose the encoding such that for all $j \in [1..n]$ with probability $1/b$ the j^{th} word of the encoding is the i^{th} word of the message, i.e., $t = 1$.

This method contains some ideas and features of the probabilistic erasure code eventually developed. For example, the expected number of encoding words necessary to get the i^{th} message word is b. However, it has several flaws including the following two related drawbacks. With probability $(1 - 1/b)^{(1+\delta)b} \approx \exp(-(1+\delta))$ more than $(1+\delta)b$ encoding words are necessary to get the i^{th} message word. Hence the variance is high and the probability of not getting the i^{th}

message word drops to $1/b$ only after $\Omega(b \log b)$ encoding words have been received.

Secondly, the case that all message words are received corresponds exactly to the classical coupon collecting problem. Hence the expected number of encoding words necessary to receive all message words is $\Omega(b \log b)$, i.e., the encoding must have length $\Omega(m \log m)$ instead of linear length.

To overcome these problems we combine this method with erasure codes. Let $t > 1$ be the size of a bundle and let $U_1, \ldots, U_{b/t}$, be the partition of the message M into bundles. A bundle U_i is viewed as the coefficients of a polynomial G_i of degree $t-1$ over GF$[2^w]$. The r^{th} word of the code E is chosen as follows. An index $i_r \in \{1, \ldots, b/t\}$ and an element $s_r \in$ GF$[2^w]$ are chosen uniformly at random, and code word E_r is set to $G_{i_r}(s_r)$.

Given a set of $(1+\delta)b \leq n$ words of E, a fixed bundle U_i can be recovered from this set if it contains the value of G_i at t different elements of GF$[2^w]$. Using Chernoff-bounds (e.g., see [2, Alon, Spencer]), the following lemma is easy to prove.

Lemma 5.1 *Let $1 \geq \delta > 2/(\mu-2)$. For all messages M, any fixed bundle U_i and any fixed set of $(1+\delta)b \leq n$ words of the encoding, with probability at least $1 - \exp(-\delta^2 t/16)$ the bundle U_i of M can be recovered from these code words. The probability is over the random choices of the bundles and field elements.*

Choosing for each code word a bundle and a field element uniformly at random requires a lot of truly random bits. Using the construction of (γ, k)-independent random variables given in [1, AGHP], and using the analysis given in [5, EGLNV], the number of random bits required by the probabilistic erasure codes can be reduced significantly. The details of this method are described in the full paper.

Replacing deterministic erasure codes by probabilistic erasure codes, we obtain a probabilistic version of Lemma 4.2. Combining this with Lemma 4.4, we obtain the following probabilistic version of Theorem 4.3.

Theorem 5.2 *Let β be a priority function with* $\text{rate}_\beta = 1$ *for messages of length m. There is an efficient algorithm that, on input β, m, a pair of integers $\langle w, t\rangle$, satisfying $2^w \geq \mu \cdot t, \mu \geq 2$, and value $\alpha \geq 3$ produces a PET system such for each $\delta \in (2/(\mu-2), 1]$ the system has a pair of priority function/failure probability $(\gamma(\delta), p(\delta))$ with the following properties:*

(i) The encoding length e is at most $3(1+\delta)\alpha m$.

(ii) The packet size ℓ is at most $\alpha^2 \log(2\alpha m) w$.

(iii) For all $i \in [1..m]$, $\gamma_i(\delta) \leq (1+\delta)(1+5/\alpha) \cdot \beta_i$.

(iv) $\text{rate}_\gamma = 1/(1+\delta)$.

(v) The failure probability $p = p(\delta)$ satisfies

$$p(\delta) \leq \exp(-\delta^2 t/16).$$

As in Theorem 4.3 the parameter α balances the tradeoff between the closeness of the priority function γ to β, the total encoding length, and the packet size. The parameter ϵ balances the tradeoff between the total encoding length and the range for which a precise statement about the confidence function can be made. The pair of parameters $\langle w, t\rangle$ balance a tradeoff between the efficiency of the encoding and decoding processes and how fast the confidence function p decreases with increasing δ.

6 Inherent limits of PET systems

In this section we describe some of the inherent limitations of PET systems. We start with a sketch of the bound on the rate of any PET system. Let $\mathcal{E} \subseteq \{E(M) \mid M \in \{0,1\}^m\}$ denote an arbitrary subset of the encodings sent by the system. A PET encoding $E = \langle E_1, \ldots, E_n\rangle \in \left[\{0,1\}^\ell\right]^n$ has length e and is broken into $n = e/\ell$ packets of size ℓ. Hence, it can be viewed as a point in the n dimensional lattice $\mathbf{Z}^n$ where each coordinate lies between 0 and $2^\ell - 1$. The set of encodings $\mathcal{E}$ can be viewed as a set of such points.

Definition 6.1 *For each $q \in [1..n]$ we define the following measure of $\mathcal{E}$:*

$$V_q(\mathcal{E}) = \left(\prod_{\vec{t} \in \binom{n}{q}} |\mathcal{E}_{\vec{t}}| \right)^{\frac{1}{\binom{n-1}{q-1}}}.$$

Here $\mathcal{E}_{\vec{t}}$ denotes the projection of $\mathcal{E}$ onto the dimensions $\vec{t}$, where $\vec{t} \in \binom{n}{q}$ is any q of the n dimensions.

Lemma 6.2

$$a^n \geq V_1(\mathcal{E}) \geq V_2(\mathcal{E}) \geq \ldots \geq V_n(\mathcal{E}) = |\mathcal{E}|.$$

This lemma is an extension of a result given in [7, Loomis, Whitney]. A similar generalization was previously proved in [3, Ben-Or, Linial].

Definition 6.3 *Let $\mathcal{E} = \mathcal{E}^0 \dot\cup \mathcal{E}^1$ and let $b = 0$ on $\mathcal{E}^0$ and $b = 1$ on $\mathcal{E}^1$. For $\vec{t} \in \binom{n}{q}$, we say that the coordinates $\vec{t}$* determines *b if $\mathcal{E}^0_{\vec{t}} \dot\cup \mathcal{E}^1_{\vec{t}}$ is a partition of $\mathcal{E}_{\vec{t}}$.*

Lemma 6.4 *If every q coordinates determines the bit b, then there is a setting of $b \in \{0,1\}$ for which*

$$V_q(\mathcal{E}^b) \le 2^{-n/q} \cdot V_q(\mathcal{E}).$$

Theorem 6.5 *For any PET with priority function β,* $\text{rate}_\beta \le 1$.

Proof of Theorem 6.5: Let

$$\mathcal{E} \subseteq \{E(M) \mid M \in \{0,1\}^m\}$$

denote the set of encodings associated with messages sent by the PET system. For all $i \in [1..m]$, let $q_i = \beta_i/\ell$ denote the number of packets needed to determine the message bit M_i in the PET system. For any sequence of $\langle b_1 \cdots b_i \rangle$ of i bits, let $\mathcal{E}^{b_1 \cdots b_i}$ denote the set of encodings possible subject to $M_1 = b_1, \ldots, M_i = b_i$. Let $\vec{t} \in \binom{n}{q_i}$ denote any q_i of the n packets. Let $\mathcal{E}^{b_1 \cdots b_{i-1} 0} \dot\cup \mathcal{E}^{b_1 \cdots b_{i-1} 1}$ be the partition of $\mathcal{E}^{b_1 \cdots b_{i-1}}$ based on whether M_i is 0 or 1. Since the value of M_i is determined by the values of the packets in $\vec{t}$, it follows that $\mathcal{E}^{b_1 \cdots b_{i-1} 0}_{\vec{t}} \dot\cup \mathcal{E}^{b_1 \cdots b_{i-1} 1}_{\vec{t}}$ is a partition of $\mathcal{E}^{b_1 \cdots b_{i-1}}_{\vec{t}}$.

Applying Lemma 6.2 followed by Lemma 6.4 in sequence m times, and using $q_1 \le q_2 \le \ldots \le q_m$ and $n/q_i = e/\beta_i$, it follows that there is a setting $\langle b_1 \cdots b_m \rangle$ for the message M such that

$$\begin{aligned}
2^e &\ge V_{q_1}(\mathcal{E}) \\
&\ge 2^{e/\beta_1} \cdot V_{q_1}(\mathcal{E}^{b_1}) \\
&\ge 2^{e/\beta_1} \cdot V_{q_2}(\mathcal{E}^{b_1}) \\
&\ge 2^{e/\beta_2} 2^{e/\beta_1} \cdot V_{q_2}(\mathcal{E}^{b_1 b_2}) \\
&\vdots \\
&\ge 2^{\sum_{i \in [1..m]} e/\beta_i} \cdot V_{q_m}(\mathcal{E}^{b_1 b_2 \ldots b_m}) \\
&\ge 2^{\sum_{i \in [1..m]} e/\beta_i} \cdot |\mathcal{E}^{b_1 b_2 \ldots b_m}|
\end{aligned}$$

Note that $|\mathcal{E}^{b_1 b_2 \ldots b_m}| \ge 1$, because there is an encoding $E \in \mathcal{E}$ sent when the message is fixed to $M = \langle b_1 b_2 \ldots b_m \rangle$. This gives

$$e \ge \sum_{i \in [1..m]} e/\beta_i = e \cdot \text{rate}_\beta$$

□

The lower bound for the probabilistic case and its proof are similar.

Theorem 6.6 *For any probabilistic PET system with priority function β and failure probability p,* $\text{rate}_\beta \le 1/(1-p)$.

6.1 A lower bound on the packet size

Theorem 6.7 *Consider a system that encodes messages of length b into codes of length e with packet size ℓ, so that any β/ℓ of the packets determines the entire message. Then $\ell \ge \log(e/\beta) - O((e/\beta)2^{-b})$.*

There are two corollaries of Theorem 6.7. For erasure codes, $\beta = b$ and $\ell = w$, because any b/ℓ of the packets determines the entire message and the word length is w. Therefore, for any erasure code (not just those using polynomials), the word length w must be at least $\log(e/b) - O((e/b)2^{-b})$.

Secondly, consider a priority function β with the property that $e^{1-\epsilon}$ bits of the code determines at least $\log(e)$ bits of the message, i.e. $\beta_{\log(e)} \le e^{1-\epsilon}$. Theorem 6.7 implies that a PET system with such a priority function β requires packets of size at least

$$\ell \ge \log(e/\beta) - O((e/\beta)2^{-b}) \ge \epsilon \log(e) - 1.$$

Acknowledgment: We thank Celina Albanese for helping to clarify the presentation in this paper. We also thank Richard Karp for help in the proofs of Section 6.

References

[1] N. Alon, O. Goldreich, J. Hastad, R. Peralta, *Simple constructions of almost k-wise independent random variables*, Random Structures and Algorithms, 3(3) (1992), pp. 289-304.

[2] N. Alon, J. H. Spencer, *The probabilistic method*, John Wiley & Sons, Inc., New York, 1992.

[3] M. Ben-Or, N. Linial, *Collective coin flipping*, Randomness and Computation, S. Micali ed., Academic Press, New York, 1989, pp. 91-115.

[4] E. W. Biersack, *Performance evaluation of forward error correction in an ATM enviroment*, Journal of Selected Areas in Communication, 11(2)(1993), pp. 631-640.

[5] G. Even, O. Goldreich, M. Luby, N. Nisan, B. Veličković, *Approximations of general independent distributions*, in Proc. 24^{th} Symposium on Theory of Computing (STOC), 1992, pp. 10-16.

[6] D. Le Gall, *MPEG: A Video Compression Standard for Multimedia Applications*, CACM, Vol. 34, No. 4, April 1991, pp. 47-58.

[7] L. H. Loomis, H. Whitney, *An inequality related to the isoperimetric inequality*, Bulletin of the American Mathematical Society, 55(7) (1949), pp. 961-962.

[8] A. J. McAuley, *Reliable broadband communication using a burst erasure correcting code*, in Proceedings SIGCOMM'90, Philadelphia, 1990.

[9] M. O. Rabin, *Efficient Dispersal of Information for Security, Load Balancing, and Fault Tolerance*, J. ACM, Vol. 36, No. 2, April 1989, pp. 335-348.

[10] N. Shacham, *Multicast Routing of Hierarchical Data,* Proceedings of ICC'92, Chicago, 1992.

[11] G. K. Wallace, *The JPEG Still Picture Compression Standard*, CACM, Vol. 34, No. 4, April 1991, pp. 30-44.