Priority Encoding Transmission

Andres Albanese   Johannes Blömer*

Jeff Edmonds†   Michael Luby‡

TR-94-039

August 1994

International Computer Science Institute
1947 Center Street
Berkeley, CA 94704
e-mail: {aa, bloemer, edmonds, luby}@icsi.berkeley.edu

*Partially supported by NSF operating grant CCR-9304722, and ESPRIT BR Grant EC-US 030.

†Supported by an NSF postdoctoral fellowship and by a Canadian NSERC postdoctoral fellowship.

‡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.

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 immediate applications to multi-media and high speed networks applications, especially in those with bursty sources and multiple receivers with heterogeneous capabilities. Implementations of the system show promise of being practical.

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 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 [9, McAuley] and empirically evaluated by [4, Biersack]. A similar encoding system has been proposed by [10, 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.

[11, 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 is an analysis of a proposal for using PET to send IP datagrams over ATM networks. Section 7 defines a geometric measure of information and gives an information-theoretic proof that the rate of any PET system is at most 1.

Little effort is made to make the systems efficient. A subsequent paper will concentrate on efficient implementations of PET systems.

2 Applications

Priority Encoding Transmission is a new method for sending information messages through an 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 [12, Wallace], [7, 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 Definition of a PET system

The basic setup for the models is a medium in which information has to be transmitted. Information is transmitted in form of packets of bits. These packets are the basic unit of information.

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 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

In this system the encoding and decoding is done deterministically. A guarantee is given that once a certain number of bits of the encoding is received the decoding of certain pieces of the message is always successful. 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 packet 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/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 $\text{rate}_\beta \leq 1$.

Theorem 3.3 *For any PET system its priority function β satisfies*

$$\text{rate}_\beta \leq 1.$$

A formal proof of this theorem is given in Section 7.

It will also be shown (Theorem 4.2) 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 $i \in [1..m]$

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

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.

Instead of specifying for each message bit the number of bits needed to recover it from the encoding it may sometimes be more useful to specify for each message bit the percentage of the encoding needed to recover it. Hence *the fractional form of a priority function* is a function ρ that maps $[1..m]$ to integer multiples of $1/n$, where n is the number of packets sent. The meaning is that message bit M_i should be recoverable from any ρ_i fraction of the encoding. The values of ρ are given in multiples of $1/n$ since we assume that only complete packets of the encoding are received. If β is a priority function and ρ its fractional form then

$$\rho_i = \frac{\beta_i}{e} \text{ and } \beta_i = \rho_i e. \tag{1}$$

Hence, if ρ is the fractional form of a priority function of a PET system then the rate of the system is

$$\sum_{i \in [1..m]} \frac{1}{\rho_i e}.$$

Therefore Theorem 3.3 implies

Theorem 3.4 *For a PET system with message length m, packet size s and priority function ρ in fractional form the encoding length $e = n\ell$ satisfies*

$$e \geq \sum_{i \in [1..m]} \frac{1}{\rho_i}.$$

It follows immediately from Equation 3.1 that a PET system that achieves rate one also achieves an optimal encoding length.

There is crucial difference between Theorem 3.4 and Theorem 3.3. Theorem 3.3 states that even if there is no bound on the length of the encoding no PET system exists for certain priority functions as defined in Definition 3.1.On the other hand, Theorem 3.4 only gives a lower bound on the encoding length necessary to implement a priority function in fractional form. This reflects the fact that as defined in Definition 3.1 the priority function is an absolute measure for the priority of a bit while the fractional form of a priority functions is a relative measure of priority.

3.2 The probabilistic model

In the model described in this section the encoding and decoding is done via randomized algorithms. Unlike in the previous model the decoding guarantee is only with high probability.

Definition 3.5 (probabilistic PET system) *A probabilistic PET system with message length m, packet size ℓ, n packets, encoding length $e = n\ell$, failure probability $p > 0$ and using r random bits consists of the following:*

(i) A family of encoding functions E^R, $R \in \{0,1\}^r$, that map a message M of length m onto an encoding consisting of n packets, i.e. $n\ell$ bits.

(ii) A family of decoding functions D^R, $R \in \{0,1\}^r$, that map a set of at most n packets onto a bit string of length m.

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

The guarantee of the system 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 $R \in \{0,1\}^r$.

In the probabilistic model it is assumed that a *common* random string R is used for the encoding and the decoding. Once the string R has been selected the encoding and decoding is deterministic. We stress that the failure probability is not over a particular distribution over the messages. For any fixed value of R an encoding/decoding pair E^R, D^R succeeds or fails on certain subsets of packets, independent of the message.

The priority function β has a similar meaning as in the deterministic model, except that even if more than β_i/ℓ packets are received there is a (small) chance that the decoding function fails to decode the i-th message bit.

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.

In Section 7 we show that Theorem 3.3 can be generalized to probabilistic PET systems in the following way.

Theorem 3.6 *If a probabilistic PET system achieves a failure probability p then its rate is at most $1/(1-p)$.*

Analogous to deterministic PET systems the priority function of a probabilistic PET system can also be expressed in fractional form.

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.

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 $\mathrm{GF}[2^w]$. Call this polynomial G. The j^{th} word of the code consists of the value of the polynomial G evaluated at the field element $j \in \mathrm{GF}[2^w]$. 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

$$w \geq \log(n). \tag{2}$$

This ensures that there are at least n different elements in the field $\mathrm{GF}[2^w]$ on which to evaluate the polynomial. Theorem 7.9 found in Section 7 proves that for any erasure code (not just those using polynomials), the word length w must be at least $\log(n/b)$.

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.

An *m-partition* consists of a sequence of positive integers

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

such that

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

Let M be a message of length m, and let $B_1, \ldots, B_d$ be the d blocks of M with respective lengths $m_1, \ldots, m_d$. An *m-partition to packet mapping* describes at a high level how to implement a PET system based on an m-partition. A packet consists of d equal length words, and the mapping associates the j^{th} block of the message with the j^{th} word of the packet. The PET system puts information about block B_j in the j^{th} word of each packet. In the implementations described below of PET systems based on this mapping, it turns out that all bits in the same block B_j have the same priority value $\gamma_j = dm_j$.

Given a partition of the message into blocks, the next lemma shows how to construct a PET system based on the mapping and on erasure codes. In this and all subsequent constructions, we ignore small roundoff errors.

Lemma 4.1 *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) $\mathrm{rate}_\gamma = 1$.

(iii) *The total 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 4.1: 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

$$\gamma_j = \ell b_j = dm_j. \tag{3}$$

This proves item (i). Note that

$$\mathrm{rate}_\gamma = \sum_{j \in [1..d]} m_j/\gamma_j = 1. \tag{4}$$

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

$$n \geq \max_{j \in [1..d]}\{b_j\}. \tag{5}$$

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 (2). Thus, we can set $w = \log(e) \geq \log(n)$. This proves item (iv). $\square$

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 because of the convention stated in Section 3. 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 $\mathrm{rate}_\beta = 1$, and a parameter $\alpha \geq 3$. When this m-partition is used to construct a PET system as described in Lemma 4.1, 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 state the main theorem (Theorem 4.2) and the partitioning lemma (Lemma 4.3) upon which the theorem is based. We then prove Theorem 4.2, which follows from a straightforward combination of Lemma 4.3 and Lemma 4.1. Before proving Lemma 4.3 we give some examples of particular priority functions and show how to partition the message to implement them.

Theorem 4.2 *Let β be a priority function with $\mathrm{rate}_\beta = 1$ 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) $\mathrm{rate}_\gamma = 1$.

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.

Lemma 4.3 *Let β be a priority function with $\mathrm{rate}_\beta = 1$ 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, $m_j \le (1+5/\alpha)\cdot \beta_i/d$.

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

Proof of Theorem 4.2: The first step is to partition M based on β, m and α as described in Lemma 4.3. We then use Lemma 4.1 to get the PET system. It is easy to verify that it has claimed properties. ◻

In practice, the transmission medium dictates the size ℓ_{medium} of a packet. If $\ell > \ell_{\text{medium}}$ then the PET system cannot be implemented as described on this medium. Even for the ATM standard, where the payload of a packet is rather small, i.e., $\ell_{\text{medium}} = 48$ bytes, this limitation is unlikely to be of great concern. On the other hand, if $\ell < \ell_{\text{medium}}$ then the packet size of the PET system can be easily scaled up to ℓ_{medium} so that the total encoding length, the priority function and the fractional form of the priority function all remain unchanged.

We now give three examples priority functions β and show how to directly implement them using a block system. For all the examples, the word length is $w \approx \log(m)$, the number of words in a packet is $d \approx \log(m)$ and thus the length of a packet is $\ell \approx \log^2(m)$. Furthermore, unlike in Theorem 4.2 the total encoding length is $e_{\min} = \gamma_d \approx m\log(m)$, which is more than linear in m. The reason for this is that, because all the priority functions grow essentially linearly with the message index, the number m_d of message bits in the last message block B_d is a constant fraction of m, and $\gamma_d = dm_d$ from Equation (3). Counter intuitively, the construction given in Lemma 4.3 to prove Theorem 4.2 actually makes the priority for the last bits higher than required by β to achieve a linear total encoding length.

Example 1: This is an example where the priority of a bit grows linearly with its index, i.e., $\beta_i \approx i\log(m)$. For $j \in [1..d]$, let B_j be a block of size $m_j = 2^j$, and thus from Equation (3) the bit priority of all bits in B_j is $\gamma_j = d2^j$.

Example 2: This is an example where earlier bits are given higher priority than was the case in Example 1, i.e., $\beta_i \approx i\log^2(i)$. For $j \in [1..d]$, let B_j be a block of size $m_j = j^2 2^j/d$, and thus from Equation (3) the bit priority of all bits in B_j is $\gamma_j = j^2 2^j$.

Example 3: This is an example where later bits are given higher priority than was the case in Example 1, i.e., $\beta_i \approx i\log^2(m/i)$. For $j \in [1..d]$, let B_j be a block of size $m_j = (d-j)^2 2^j/d$, and thus from Equation (3) the bit priority of all bits in B_j is $\gamma_j = (d-j)^2 2^j$.

Proof of Lemma 4.3: 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 $1 \le c' \le 1 + 1/\alpha$ is a small normalizing constant that makes $\text{rate}_{\beta'} = 1$. Note that for all i,

$$\beta'_i \le c'\beta_i. \tag{6}$$

Furthermore,

$$\beta'_m \le c'\alpha m \le 2\alpha m. \tag{7}$$

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:

$$c^{k'} \geq 2\alpha m. \tag{8}$$

$$k = d(1 - 1/\alpha). \tag{9}$$

Inequality (8) 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 (7), and from Inequality (8), 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 (6), it follows that, for all indices i in block B_j,

$$m_j \leq c \cdot (d/k) \cdot \beta'_{i_j}/d \leq c \cdot (d/k) \cdot \beta'_i/d \leq c' \cdot c \cdot (d/k) \cdot \beta_i/d. \tag{10}$$

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 $\alpha \geq 3$,

$$(1 + 1/\alpha)^2/(1 - 1/\alpha) \leq 1 + 5/\alpha. \tag{11}$$

From this inequality, and from Inequality (10), 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 (10), and because $\beta'_{i_j} \leq \beta'_m \leq c'\alpha m$ from Inequality (7), it follows that

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

Thus, from Inequality (11), 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

In this section we describe probabilistic PET systems. The main building blocks in these system are probabilistic erasure codes. The basic ideas for these codes are presented in the first subsection. A detailed description for probabilistic erasure codes is given in the second subsection. The third subsection shows how to reduce the number of truly random bits used in these probabilistic erasure codes. Based on these probabilistic erasure codes in last subsection we describe probabilistic PET systems.

5.1 Basic ideas

In this section we present the basic primitive for the probabilistic PET system.

Erasure codes as described in Section 4.1 and used in the deterministic PET system in Section 4.3 are specified by a triple $\langle b, n, w \rangle$ such that recovering the message M of length $m = wb$ requires the interpolation of polynomial of degree $b-1$ over GF$[2^w]$. For large values of b and w this may turn out to be infeasible. In the following two sections we present a probabilistic erasure code that allows a smaller word size and smaller degree polynomials. The basic 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 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.

5.2 The probabilistic erasure code using truly random bits

A probabilistic erasure code is specified by a 4-tuple $\langle b, t, n, w \rangle$, where $n \geq b \geq t$. It encodes a message M of length wb into a code $E(M)$ of length wn. The message and the code consist of words of length w. The message is furthermore broken into b/t bundles $U_1, \ldots, U_{b/t}$, each containing t words.

A word is viewed as an element in $\mathrm{GF}[2^w]$ and the t words in a bundle are viewed as the coefficients of a polynomial of degree $t-1$ over $\mathrm{GF}[2^w]$. Let G_i be the polynomial described by the i^{th} bundle.

For later purposes the encoding of a message M of length wb is conveniently described by introducing n mutually independent and identically distributed $(b/t \cdot 2^w)$-valued random variables $X_1, \ldots, X_n$, i.e., for all $j \in [1..n]$ and all pairs $(i,s) \in [1..b/t] \times \mathrm{GF}[2^w]$

$$\Pr[X_j = (i,s)] = t/b \cdot 2^{-w}.$$

We now define the encoding of message M in terms of $X_1, \ldots, X_n$, i.e. if $X_j = (i,s)$ then the j^{th} word of the code consists of the value $G_i(s)$.

If $b/t = 2^r$, the distribution on the random variables X_j can be generated using a random string $R \in \{0,1\}^{n(r+w)}$. If b/t is not a power of 2 and a random string $R \in \{0,1\}^{n(r+w)}$ is used than 2^r should be large compared to b/t. In this situation certain bundles a more likely to be chosen. However, by choosing r large enough this difference can be made minuscule and in the sequel we will ignore this roundoff-error.

Next we analyze the probability that given the random string R and a fixed set of $s = (1+\delta)b \leq n$ words of $E(M)$ a fixed bundle U_i cannot be recovered. The probability is with respect to the random string R and independent of the message M.

Without loss of generality assume that the set of code words consists of the first $s = (1+\delta)b$ words of $E(M)$. The decoding of bundle U_i is impossible if the set of code words contains the value of G_i at less than t different elements of $\mathrm{GF}[2^w]$. Therefore let $\mu = 2^w/t \geq 1$ be the factor by which the field size is larger than the degree of the polynomial. The failure probability for decoding will depend on μ as well as on t and δ.

Let us call the j^{th} word of the code *successful* if it is about the i^{th} bundle and if either the value of G_i at t different elements of $\mathrm{GF}[2^w]$ is already known from the first $j-1$ code words, or if the j^{th} code word gives the value of G_i at a new element of $\mathrm{GF}[2^w]$. The i^{th} bundle can be recovered from the first u encoding words if at least t of these words are successful.

Independent of the previous words the j^{th} code word is successful with probability at least

$$\frac{t}{b}\left(1 - \frac{1}{\mu}\right).$$

More formally, for the j^{th} encoding word we introduce mutually independent indicator random variable $Z_j, j = 1, \ldots, s$, such that $Z_j = 1$ if the j^{th} word is successful. Hence

$$\Pr[Z_j = 1] \geq \frac{t}{b}\left(1 - \frac{1}{\mu}\right), \; j = 1, \ldots, s.$$

and the number of successful words among the first u words of the encoding is $Z = \sum_{j=1}^{s} Z_j$. Furthermore, the probability that the i^{th} bundle cannot be recovered from the first s words is $\Pr[Z < t]$.

$$\text{Exp}[Z] = \sum_{j=1}^{s} \Pr[Z_j = 1] \geq (1+\delta)(1-1/\mu)t.$$

We want $\text{Exp}[Z] \geq t$ and thus $(1+\delta)(1-1/\mu) \leq 1$ or, equivalently, $\delta \leq 1/(\mu-1)$. Given that $\text{Exp}[Z] \geq t$ we can use Chernoff-bounds (see for example [2]) to bound $\Pr([Z < t])$:

$$\Pr[Z < t] \leq \Pr[Z - \text{Exp}[Z] < -(\delta(1-1/\mu) - 1/\mu))t] =$$

$$= \Pr\left[Z - \text{Exp}[Z] < -\frac{(\delta(1-1/\mu) - 1/\mu)}{(1+\delta)(1-1/\mu)}\text{Exp}[Z]\right] \leq \mathrm{e}^{-\frac{(\delta(1-1/\mu)-1/\mu)^2 t}{2(1+\delta)(1-1/\mu)}}.$$

To derive a simple upper bound for this expression observe first that for $\delta \leq 1$ the denominator $2(1+\delta)(1-1/\mu)$ of the exponent is less than 4. Next, $(\delta(1-1/\mu)-1/\mu)^2 t$ can be written as $(\delta - (\delta+1)/\mu)^2\, t$. If $\delta \geq 2/(\mu-2)$ then $(\delta+1)/\mu \leq 1/2\delta$ and $(\delta-(\delta+1)/\mu)^2 \geq \delta^2/4$. This proves

Lemma 5.1 *Let $1 \geq \delta \geq 2/(\mu-2)$. For all messages M, any fixed bundle U_i and any fixed set of $(1+\delta)b \leq n$ words of the encoding $E^R(M)$, with probability at least*

$$1 - \mathrm{e}^{-\frac{\delta^2}{16}t}$$

the bundle U_i of M can be recovered from the random string $R \in \{0,1\}^{n(r+w)}$ and the set of words. The probability is over the uniform distribution of the random string $R \in \{0,1\}^{n(r+w)}$.

5.3 Erasure code based system using fewer random bits

The purpose of this section is to show how to reduce the number of truly random bits used in the probabilistic erasure code. The basis idea is to replace the mutually independent random variables X_j defined on page 13 by k-wise independent random variables that approximate the random variables X_j.

Definition 5.2 **((k,γ)-approximations for independent random variables)** *Let $X_i, i = 1,\ldots,n$, and $Y_i, i = 1,\ldots,n$, be random variables that take on values in $[0..m-1]$. Assume the X_i's are mutually independent. The random variables Y_i are called a (k,γ)-approximation to the random variables $X_1,\ldots,X_n$, if for any $l \leq k$, any l distinct indices $1 \leq i_1 < \ldots < i_l \leq n$ and any string $(v_1,\ldots,v_l) \in [0..m-1]^l$*

$$\left|\Pr[Y_{i_1} = v_1,\ldots,Y_{i_l} = v_l] - \prod_{j=1}^{l} \Pr[X_{i_j} = v_j]\right| \leq \gamma.$$

(k,γ)-approximations to n independent m-valued random variables can be generated by constructing small sample spaces $S \subset [0..m-1]^n$. S induces a distribution on n random variables by choosing a point randomly and uniformly from S. The value of the j^{th} random variable is determined by the j^{th} coordinate of the element in S.

In [1] constructions for sample spaces that induce (k,γ)-approximations to $\mathcal{U}_{n,2}$ i.e., n identically and uniformly distributed boolean-valued random variables, have been given. These sample spaces are of size

$$\left(\frac{k \log n}{2\gamma}\right)^2 .$$

Based on this result in [5] a sample space was defined that induces a (k,γ)-approximation to the distribution on n identically and uniformly distributed m-valued random variables.

Assuming that $2k \leq \log 1/\gamma$ the size of this sample space is

$$\left(\frac{7 \log \frac{1}{\gamma} \cdot \log\left(7n \log \frac{1}{\gamma}\right)}{\gamma^8}\right)^2 .$$

In particular, this construction can be used to obtain a (k,γ) approximation to the mutually independent random variables X_j defined on page 13.

If this approximation to the variables X_j is used then the number of successful words for a bundle U_i of the message M is given by the sum of random variables that are a $\left(k, ((b/t)2^w)^k \gamma\right)$-approximation to the indicator random variables Z_j defined on page 13. To see this observe that any event of the form $(Z_{i_1} = \epsilon_1, \ldots, Z_{i_k} = \epsilon_k), \epsilon_{i_j} \in \{0,1\}$, is the union of at most $((b/t)2^w)^k$ events of the form $(X_{i_1} = \alpha_1, \ldots, X_{i_k} = \alpha_k), \alpha_{i_j} \in [1..b/t] \times \mathrm{GF}[2^w]$.

The following theorem will enable us to bound the failure probability for decoding a fixed bundle if (k,γ)-approximations to mutually independent random variables are used for the probabilistic erasure codes.

Theorem 5.3 *Let $X_1, \ldots, X_u$ be mutually independent $0,1$-random variables with $\Pr[X_i = 1] = p_i$. Set $p = \sum_{i \in [1..u]} p_i/u$. Hence $\mathrm{Exp}[\sum X_i] = up$. Denote $X_i - \mathrm{Exp}[X_i] = X_i - p_i$ by $\overline{X}_i$. Let $Y_1, \ldots, Y_u$ be (k,γ)-approximations to the random variables X_i, where $k = up/e$ and $\gamma = \exp(-up \ln(4/p)/\mathrm{e})$. Set $\overline{Y}_i = Y_i - p_i$, $\overline{Y} = \sum_{i \in [1..u]} \overline{Y}_i$. For any $\epsilon \in (0,1]$*

$$\Pr[|\overline{Y}| > \epsilon up] < 3 \cdot 2^{-\epsilon^2 up/2\mathrm{e}}.$$

Proof: Let $l \leq k$ be some even integer whose value will be determined later. By Markov's inequality and the triangle inequality

$$\Pr[|\overline{Y}| > \epsilon up] \leq \left(\frac{1}{\epsilon up}\right)^l \mathrm{Exp}[\overline{Y}^l] \leq$$

$$\left(\frac{1}{\epsilon up}\right)^l \left(\sum_{1 \leq i_1, \ldots, i_l \leq u} \left|\mathrm{Exp}[\overline{Y}_{i_1} \cdots \overline{Y}_{i_l}] - \mathrm{Exp}[\overline{X}_{i_1} \cdots \overline{X}_{i_l}]\right|\right)$$

$$+ \left(\frac{1}{\epsilon up}\right)^l \left(\sum_{1 \leq i_1, \ldots, i_l \leq u} \mathrm{Exp}[\overline{X}_{i_1} \cdots \overline{X}_{i_l}]\right) .$$

The fact that the Y_i's are (k,γ)-approximations to the X_i's implies that for any of the u^l l-tuple $i_1,\ldots,i_l$ and for any vector $(v_1,\ldots,v_l)\in\{0,1\}^l$

$$|\Pr[\overline{Y}_{i_1}=v_1,\cdots\overline{Y}_{i_l}=v_l]-\Pr[\overline{X}_{i_1}=v_1\cdots\overline{X}_{i_l}=v_l]|\le\gamma.$$

Hence

$$\left(\frac{1}{\epsilon up}\right)^l\left(\sum_{1\le i_1,\ldots,i_l\le u}\left|\mathrm{Exp}[\overline{Y}_{i_1}\cdots\overline{Y}_{i_l}]-\mathrm{Exp}[\overline{X}_{i_1}\cdots\overline{X}_{i_l}]\right|\right)\le\left(\frac{1}{\epsilon up}\right)^l(2u)^l\gamma.$$

Set

$$l=\epsilon^2up/\mathrm{e}\le k.$$

This gives $(\epsilon up)^{-l}(2u)^l\gamma=(2/\epsilon p)^l\mathrm{e}^{-up\ln(4/p)/\mathrm{e}}$. Using derivatives it can be shown that for any $\epsilon\in(0,1]$

$$\mathrm{e}^{-up\ln(4/p)/\mathrm{e}}\le\mathrm{e}^{-\epsilon^2up\ln(4/p\epsilon)/\mathrm{e}}=\left(\frac{\epsilon p}{4}\right)^{\epsilon^2up/e}.$$

Hence

$$(2/\epsilon p)^l\mathrm{e}^{-up\ln(4/p)/\mathrm{e}}\le(2/\epsilon p)^{\epsilon^2up/\mathrm{e}}(\epsilon p/4)^{\epsilon^2up/\mathrm{e}}\le 2^{-\epsilon^2up/\mathrm{e}}.$$

Next consider

$$\sum_{1\le i_1,\ldots,i_l\le u}\mathrm{Exp}[\overline{X}_{i_1}\cdots\overline{X}_{i_l}].$$

Since the random variables $\overline{X_i}$ are mutually independent this sum can be rearranged to

$$\sum_{t=1}^{l/2}\sum_{\substack{1\le i_1<i_2<\ldots<i_t\le u\\ e_1,\ldots,e_t\\ \sum e_j=l,\,e_j\ge1}}\binom{l}{e_1\ldots e_t}\mathrm{Exp}[\overline{X}_{i_1}^{e_1}]\cdots\mathrm{Exp}[\overline{X}_{i_t}^{e_t}].$$

Observe that for all $\overline{X_i}$ the expectation $\mathrm{Exp}[\overline{X_i}]=0$ and that for any integer $e\ge2$

$$\mathrm{Exp}[\overline{X}_i^e]\le p_{i_j}=\mathrm{Exp}[X_i].$$

Combining this with

$$\sum_{\substack{e_1,\ldots,e_l\\ \sum e_j=l}}\binom{k}{e_1\ldots e_t}=t^l$$

yields

$$\sum_{\substack{1\le i_1<i_2<\ldots<i_t\le u\\ e_1,\ldots,e_t\\ \sum e_j=l,\,e_j\ge1}}\binom{l}{e_1\ldots e_t}\mathrm{Exp}[\overline{X}_{i_1}^{e_1}]\cdots\mathrm{Exp}[\overline{X}_{i_t}^{e_t}]=$$

$$\sum_{\substack{1\le i_1<i_2<\ldots<i_t\le u\\ e_1,\ldots,e_t\\ \sum e_j=l,\,e_j\ge2}}\binom{l}{e_1\ldots e_t}\mathrm{Exp}[\overline{X}_{i_1}^{e_1}]\cdots\mathrm{Exp}[\overline{X}_{i_t}^{e_t}]\le$$

$$t^l \sum_{1 \le i_1 < \ldots < i_t \le u} p_{i_1} \cdots p_{i_t} \le \frac{t^l}{t!} \left(\sum_{i=1}^{u} p_i \right)^t \le t^l \left(\frac{\mathrm{e} up}{t} \right)^t,$$

where the last inequality follows from Stirling's formula.

This shows

$$\left(\frac{1}{\epsilon up} \right)^l \sum_{1 \le i_1 \le \ldots \le i_l \le u} \mathrm{Exp}[\overline{X}_{i_1} \cdots \overline{X}_{i_l}] \le \sum_{t=1}^{l/2} \frac{\mathrm{e}^t (up)^t t^l}{(\epsilon up)^l t^t}.$$

For $l = \epsilon^2 up/e$ this yields

$$\sum_{t=1}^{l/2} \frac{\mathrm{e}^t (up)^t t^l}{(\epsilon up)^l t^t} = \sum_{t=1}^{l/2} \left(\frac{\mathrm{e}}{\epsilon} \right)^t \left(\frac{t}{\epsilon up} \right)^{l-t} \le$$

$$\le \sum_{t=1}^{l/2} \left(\frac{\mathrm{e}}{\epsilon} \right)^t \left(\frac{\epsilon}{2\mathrm{e}} \right)^{l-t} = \left(\frac{\epsilon}{2} \right)^l \sum_{t=1}^{l/2} \left(\frac{2}{\epsilon^2} \right)^t \le 2 \cdot 2^{-\epsilon^2 up/2\mathrm{e}}.$$

Together with the previous estimate this proves

$$\Pr[|\overline{Y}| > \epsilon up] < 3 \cdot 2^{-\epsilon^2 up/2\mathrm{e}}.$$

□

The failure probability for a probabilistic erasure code that uses (k, γ)-approximations to mutually independent random variables is now easily analyzed.

Lemma 5.4 *Let the string R used in the probabilistic erasure code be chosen uniformly at random from a sample space that induces a (k, γ)-approximation to the distribution on n mutually independent and $b/t \cdot 2^w$-valued random variables, where*

$$k = t, \;\; \gamma = \mathrm{e}^{-2t \ln(\mu b)}.$$

Assume $1 \ge \delta > 2/(\mu - 2)$. For any message M, any fixed bundle U_i and any set of $(1 + \delta)b \le n$ words of the encoding $E^R(M)$ with probability at least

$$1 - 3 \cdot 2^{-\frac{\delta^2}{16\mathrm{e}} t}$$

the bundle U_i of M can be recovered from R and the set of code words.

Proof: Assume the probabilistic erasure code was based on a random string R that induces a (k, γ)-approximation to the distribution on n mutually independent and $b/t \cdot 2^w$-valued random variables. Without loss of generality assume that the first $(1 + \delta)b$ code words are received. We need to bound the probability that a fixed bundle U_i is not recoverable from these code words. Using our previous terminology, this in turn happens if there are less than t successful words about bundle U_i among the first $(1 + \delta)b$ code words.

As mentioned above the number of successful words about the bundle U_i is given by the random variable $T = \sum_{j=1}^{(1+\delta)b} T_j$, where the T_j's are a (k, γ')-approximation, $\gamma' = ((b/t)2^w)^t\gamma$, to the mutually independent $0,1$-random variables Z_j defined on page 13.

Hence bundle U_i is not recoverable if $T < t$. To bound the probability for this event observe that

$$\Pr[T < t] = \Pr[|T - (1+\delta)(1-1/\mu)t| > (\delta(1-1/\mu) - 1/\mu)t].$$

Since $\mathrm{Exp}[\sum Z_j] \geq (1+\delta)(1-1/\mu)t$ we can bound the latter probability by applying Theorem 5.3 with

$$\epsilon = \frac{\delta(1-1/\mu) - 1/\mu}{(1+\delta)(1-1/\mu)},$$

provided the parameters k, γ' satisfy the conditions of this theorem.

For the parameter k we have to show that $(1+\delta)(1-\mu)t/\mathrm{e} \leq t$. But this easily follows since $\delta \leq 1$ and therefore $(1+\delta)(1-1/\mu) \leq 2 \leq \mathrm{e}$.

For γ' we get

$$\gamma' = ((b/t)2^w)^t\gamma < \exp(-t\ln(8b/t) \leq \exp\left(-\frac{(1+\delta)(1-/\mu)t}{\mathrm{e}}\ln\left(\frac{4\mu b}{(mu-1)t}\right)\right).$$

Hence the condition of Theorem 5.3 is also satisfied for this parameter and the theorem yields that

$$\Pr[T < t] < 2^{-\frac{\epsilon^2(1+\delta(1-1/\mu)}{2\mathrm{e}}t}.$$

Now

$$\frac{\epsilon^2(1+\delta)(1-1/\mu)}{2\mathrm{e}} = \frac{(\delta(1-1/\mu) - 1/\mu)^2}{2\mathrm{e}(1+\delta)(1-1/\mu)} \geq \frac{(\delta - (1+\delta)^2)/\mu}{4\mathrm{e}}.$$

By choice of δ the expression $\delta - (1+\delta)/\mu$ is greater than $\delta/2$ and

$$\Pr[T < t] < 3 \cdot 2^{-\frac{\delta^2}{16\mathrm{e}}t},$$

which proves the lemma. $\square$

Using the sample space constructed in [EGLNV] and the bound for the size of this sample space we gave on page 15 it follows that choosing a random string R as described in Lemma 5.4 requires

$$32t\ln(\mu b) + 2\log(14t\ln(\mu b)) + 2\log\log(14nt\ln(\mu b)) \leq 33t\ln(\mu b) + \log\log n$$

random bits. This compares favorably to the $n(r+w)$ random bits required if R is chosen uniformly at random from $\{0,1\}^{n(r+w)}$.

5.4 Probabilistic PET systems

Using the probabilistic erasure codes described in the previous sections we now describe probabilistic PET systems. The first lemma is the probabilistic version of Lemma 4.1.

Lemma 5.5 *Given an m-partition $\langle m_1, \ldots, m_d \rangle$, for any pair of integers $\langle w, t \rangle$ satisfying $2^w = \mu \cdot t, \mu > 2$, a PET system can be constructed such that for any $\delta \in (2/(\mu-2), 1]$ the system has a pair of priority function/failure probability $(\gamma(\delta), p(\delta))$, with the following properties:*

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

(ii) $\mathrm{rate}_\gamma = 1/(1+\delta)$.

(iii) *The packet size is $d \cdot w$.*

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

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

$$p(\delta) \le 3 \cdot 2^{-\frac{\delta^2}{16e}t}.$$

(vi) *The length of the random string R is less than*

$$33t \max_{j \in [1..d]}\{\ln(\mu m_j/w) + \log\log(m_j/w)\}.$$

Proof: Fix some $\delta \in [2/(\mu-2), 1]$. For different values of δ the PET system will only differ only in the number of packets sent.

Let $B_1, \ldots, B_d$ be the blocks of M, and thus the length of B_j is m_j. Let $b_j = m_j/w$ and $n = (1+\delta)\max_{j \in [1..d]}\{b_j\}$. The probabilistic erasure code with parameters $\langle b_j, t, n, w \rangle$ is used to encode B_j into a code $E_j(B_j)$ consisting of n words. The entire encoding $E(M)$ consists of n packets of size $\ell = dw$ 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. The total encoding length is $e = \ell n = \max_{j \in [1..d]}\{\gamma_j\}$. Hence part (iii) and part (iv) of the lemma are satisfied.

Since we use a probabilistic erasure code for each block, all bits in the same block have the same priority. By Lemma 5.4 for any fixed bundle U of block B_j, given any $(1+\delta)b_j \le n$ words of the code $E_j(B_j)$ with probability $1 - p(\delta)$ the bundle U can be recovered. Since there is one such word in each packet, it follows that with probability $1 - p(\delta)$ $(1+\delta)b_j$ packets of $E(M)$ are sufficient to recover the bundle U of B_j. Since the number of bits in $(1+\delta)b_j$ packets is $(1+\delta)\ell b_j = (1+\delta)dm_j = \gamma_j(\delta)$ this proves part (i) and (v) of the lemma.

$\mathrm{rate}_\gamma = 1/(1+\delta)$ follows immediately from Lemma 4.1.

By the bound given after the proof of Lemma 5.4, for the j^{th} probabilistic erasure code a random string of length less than $33t\ln(\mu m_j/w) + \log\log(m_j/wt)$ is required. However, the random string used for the encoding and decoding of the largest block B_j can be used for each of the d probabilistic erasure codes. This proves part (vi). $\Box$

Using Lemma 5.1 instead of Lemma 5.4 we can achieve a slightly better confidence function $p(\delta)$. However, this requires a longer random string R.

Combining Lemma 5.5 with the partitioning algorithm of Lemma 4.3 yields the following probabilistic version of Theorem 4.2.

Theorem 5.6 *Let β be a priority function with* $\mathrm{rate}_\beta = 1$ *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) $\mathrm{rate}_\gamma = 1/(1+\delta)$.

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

$$p(\delta) \leq 3 \cdot 2^{-\frac{\delta^2}{16e}t}.$$

As in Theorem 4.2 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 parameters w, t balance a tradeoff between the efficiency of the encoding and decoding processes and how fast the failure probability $p(\delta)$ decreases with increasing δ.

6 Examples of PET Implementations on ATM

In this section we describe a way to implement PET systems specifically for the ATM standard, where the payload of a packet is 48 bytes. We consider two systems, a deterministic PET system for shorter messages (up to about 10M bits), and a probabilistic PET system for longer messages (from about 1M bits to about 50G bits). Both systems use words of length $w = 16$ bits or 2 bytes each. Erasure codes over the finite field $\mathrm{GF}[2^{16}]$ are used in both cases. For efficient implementation of field operations, it is useful to precompute and store in a table information about all elements in the field. The table consists of 2^{16} entries of 2 bytes each, or approximately 130K bytes. This is small enough to build the table in hardware.

Both systems are very similar in spirit. One difference between them is that for the short message system the partitioning information is specified in terms of words per block, whereas for the long message system the partitioning information is specified in terms of bundles per block.

6.1 An Example of a Short Message System for ATM

Each packet of 48 bytes is partitioned into 24 words of 2 bytes each. The message M is partitioned into 23 blocks $B_1, \ldots, B_{23}$. For $j \in [1..23]$, let T_j be the number of words in B_j. The j^{th} block B_j specifies a polynomial G_j over GF$[2^{16}]$ of degree $T_j - 1$, where each coefficient G_j consists of one word of B_j.

The last word of each packet is the identifier of the packet, i.e., it contains the field element $a \in$ GF$[2^{16}]$ at which all 23 polynomials are evaluated. For $j \in [1..23]$, the j^{th} word of a packet contains the evaluation $G_j(a)$ of the polynomial G_j at field element a. The maximum number of packets n that can be used with this system is 2^{16}, i.e., the size of the field. The priority information consists of the table $T_1, \ldots, T_{23}$. The maximum size of a block is 2^{16} words, i.e., the size of the field. Note that $\sum_{j=1}^{23} T_j$ is the number of words in the message M.

The overhead for this system (because of the use of the last word of the packet to store the identifier of the packet) is 2 bytes per packet, i.e., an additional overhead that is less than half the overhead inherent in ATM because of the 5 byte header.

We now give an example. One suggestion for high quality video using MPEG is 4Mb/second and 30 frames/second. Allowing a latency of slightly less than 0.2 seconds, we can encode 5 frames into one message, which implies that a message is 0.667M bits or 41.67K words per message. We can prioritize each frame of the MPEG message into two parts, the more important intraframe information (which we assume is 20% of the information), and the interframe information (which is the remaining 80% of the information.) We set the number of packets to $n = 2.174K$. Priority information showing what fraction of the packets are needed to recover what fraction of the prefix of the message is shown in Figure 1. The total encoding length is approximately 1.2 times the length of M, not including the overhead of 5 bytes/packet for the ATM header.

Priority Table Information		
Block Size (in words)	Fraction of Message Prefix	Fraction of Packets Needed to Recover
$T_1 = \cdots = T_6 = 1.389K$	.20	.64
$T_7 = \cdots = T_{23} = 1.961K$	1.00	.90

Figure 1: Example of Short Message

It is easy to scale up or down the number of frames per message. For example, if latency is a concern, it might be appropriate to send 1 frame per message instead of 5. In this case, $T_1, \ldots, T_{23}$ all scale down by a factor of 5, and the last two columns of Figure 1 remain the same.

Adjusting the relative priorities between the different parts of the message can be easily done simply by adjusting the relative sizes of $T_1, \ldots, T_{23}$.

6.1.1 An Example of a Long Message System for ATM

We define a bundle to consist of $t = 2^{12} = 4096$ words, or equivalently, $2^{16} \approx 65K$ bits. The intuitive reasoning for this is that we use a probabilistic PET system to implement the system, and we want to balance the two waste factors that adversely affect the priority function implemented by the system. The first waste factor is the ratio $\frac{1}{2} \cdot t/2^{16}$, which is the relative overhead for each bundle in its priority function due to randomly chosen field elements that are not distinct from previously chosen field elements. The second waste factor is $1/\sqrt{t}$, which is the variance (in relative terms) of the time it takes to choose a bundle t distinct times. Since the second waste factor is somewhat more of a concern, we set the balance so as to make the second factor somewhat smaller than the first. With our choice of t, the first factor is $\approx 3.3\%$ and the second factor is $\approx 2\%$.

Each packet of 48 bytes is partitioned into 24 words of 2 bytes each. The message M is partitioned into 22 blocks $B_1, \ldots, B_{22}$. The j^{th} block B_j consists of some number of bundles T_j. Each bundle specifies a polynomial over GF$[2^{16}]$ of degree $2^{12} - 1$, where each of the 2^{12} coefficients consist of one word from the bundle.

The identifier of each packet is the last two words. The 24^{th} word of a packet contains a value $r \in \{0,1\}^{16}$. This value is chosen randomly and is used to select one bundle from each of the 22 blocks. For all $j \in [1..22]$, let G_j be the polynomial of degree $2^{12} - 1$ corresponding to the bundle chosen from B_j. The 23^{rd} word of a packet contains a field element $a \in$ GF$[2^{16}]$. This value is also chosen randomly and it is the point at which $G_1, \ldots, G_{22}$ are evaluated. For $j \in [1..22]$, the j^{th} word of a packet contains $G_j(a)$. The maximum number of packets n that can be used with this system is unbounded. The priority information consists of the table $T_1, \ldots, T_{22}$. The maximum size of a block is 2^{16} bundles, or $2^{32} \approx 4.3G$ bits. This limitation is because the length of the random string r used to select a bundle from the block is 16 bits long. Also, because of roundoff errors in using r to choose a random bundle from the block, it is crucial that if T_j is close to 2^{16} then T_j is a power of two. In general, if $T_j = 2^{e_j}$ for some positive integer e_j then the first e_j bits of r can be used to index a random bundle from block B_j. Note that $\sum_{j=1}^{22} T_j$ is the number of bundles in the message M. The overhead for this system (because of the use of the last word of the packet to store the identifier of the packet) is 4 bytes per packet, i.e., an additional overhead that is less than the overhead inherent in ATM because of the 5 byte header.

We now give an example for a rather large message M of length 2^{30} words, i.e., approximately 17.2G bits. Note that sending such a long prioritized message using PET is not reasonable with respect to current bandwidth capacities for unicast applications. This is because the time to transmit the entire message is the time for several roundtrips, and thus it would probably be cheaper and easier to resend missing portions of the message using standard techniques. However, it may be reasonable to use PET for broadcasting large prioritized messages for multicast applications that are not real-time if there are a large number of users that want to recover different length prefixes of the message. Using PET, the sender can broadcast the encoding, and each receiver can recover as large a prefix as it wants by processing the appropriate number of packets. This may be simpler than collecting the individual quantity requests from all receivers and then making sure to send each receiver the appropriate length prefix of the message. This large message example also shows that PET can scale up to send very long messages in a rather straightforward way, and thus it will be appropriate for any very high bandwidth real-time multimedia applications of the future.

We partition the message into 22 blocks as shown in Figure 2. Thus, the lower priority blocks, i.e., blocks 9 through 22, consist of $14.99K$ bundles or $61.4M$ words each. We set the number of packets to $n = 68.2M = 61.4M/.90$. Priority information showing what fraction of the packets are needed to recover what fraction of the prefix of the message is shown in Figure 2. The total encoding length is approximately 1.52 times the length of M, not including the overhead of 5 bytes/packet for the ATM header.

Priority Table Information		
Block Size (in bundles)	Fraction of Message Prefix	Fraction of Packets Needed to Recover
$T_1 = \cdots = T_3 = 4.37K$	.05	.262
$T_4 = \cdots = T_8 = 7.86K$	.20	.472
$T_9 = \cdots = T_{22} = 14.99K$	1.000	.900

Figure 2: Example of Long Message

7 Inherent Limits for PET Systems

This section proves the following inherent limits for a PET System. For any PET system with priority function β, $\text{rate}_\beta \leq 1$. For any probabilistic PET system with failure probability p, $\text{rate}_\beta \leq 1/(1-p)$. Every PET system requires packet size of at least $\Omega(\log m)$. We begin by defining a geometric measure of information, which will be used to prove the lower bounds on the rate of a priority function.

7.1 A Geometric Measure of Information

Recall the intuition about rate_β. It states that each bit of the encoding $E(M)$ of message M contains $1/\beta_i$ bits "about" message bit M_i or equivalently the entire encoding $E(M)$ contains e/β_i bits "about" M_i and $\sum_{i \in [1..m]} e/\beta_i$ bits about M. However, the encoding is only e bits long. Hence, $\sum_{i \in [1..m]} e/\beta_i \leq e$.

The overall lower bound technique is to fix the bits of the message one at a time and to consider the set of encodings $\mathcal{E}$ that are generated from the remaining set of messages. A measure of the "size" of $\mathcal{E}$ is defined with the following property. If the encodings contain e/β_i bits "about" the message bit M_i, then M_i can be fixed in a way that decreases the measure of $\mathcal{E}$ by at least a factor of $2^{e/\beta_i}$. Initially, the measure is at most 2^e, because $\mathcal{E} \subseteq \{0,1\}^e$. Therefore, the measure of the final set $\mathcal{E}$ when all the bits of the message are fixed, is at most

$$\frac{2^e}{\prod_{i\in[1..m]} 2^{e/\beta_i}}.$$

This final measure is at least 1, because the fixed message has an encoding. It follows that $\text{rate}_\beta = \sum_{i\in[1..m]} 1/\beta_i \leq 1$.

In order to formalize this, the PET system is viewed geometrically. 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 $E(M) = \langle E_1, \ldots, E_n \rangle \in \left[\{0,1\}^\ell\right]^n$ 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.

Consider some message bit M_i. Let $\mathcal{E} = \mathcal{E}^{(0)} \dot\cup \mathcal{E}^{(1)}$ be the partition of the encodings based on M_i, i.e. for $E(M) \in \mathcal{E}^{(0)}$, $M_i = 0$ and for $E(M) \in \mathcal{E}^{(1)}$, $M_i = 1$. In a PET system, this bit M_i has the property that it is determined by any $q_i = \beta_i/\ell$ of the packets. Let $\vec{t} \in \binom{n}{q_i}$ denote any q_i of the n packets. Let $\mathcal{E}_{\vec{t}}$ denote the projection of $\mathcal{E}$ onto the q_i dimensions defined by $\vec{t}$. In the PET context, it is the set of strings that the packets $\vec{t}$ can possibly contain. Any particular string in these packets determines M_i and thus, in the geometric context, $\mathcal{E}^{(0)}_{\vec{t}} \dot\cup \mathcal{E}^{(1)}_{\vec{t}}$ is a partition of $\mathcal{E}_{\vec{t}}$.

A measure of the "size" of $\mathcal{E}$ is defined with the property that there is a way to fix M_i that decreases the measure by at least a factor of $2^{e/\beta_i} = 2^{n/q_i}$. This measure, denoted V_{q_i}, is calculated in terms of the product of the sizes of the projections $|\mathcal{E}_{\vec{t}}|$ for $\vec{t} \in \binom{n}{q_i}$.

In the following subsections we fix $a = 2^\ell$ and let $\mathcal{E} \subseteq \{0..a-1\}^n$ be an arbitrary set of points. This set will be interpreted as a subset of the lattice $\mathbf{Z}^n$. In this section, we leave the proofs of all the stated lemmas to the last subsection.

7.1.1 The Measure $V_q(\mathcal{E})$

For each $q \in [1..n]$, the measure $V_q(\mathcal{E})$ considers the projections of $\mathcal{E}$ onto all subsets of q coordinates.

Definition 7.1

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

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.

Two special cases are

$$V_1(\mathcal{E}) = \prod_{t\in[1..n]} |\mathcal{E}_t| \text{ and } V_n(\mathcal{E}) = |\mathcal{E}|.$$

The following lemma is a generalization of a theorem of Loomis and Whitney.

Lemma 7.2

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

Example: Suppose that $\mathcal{E} = S_1 \times S_2 \times \cdots \times S_n$, where $S_t \subset \mathbf{Z}$, $|S_t| = c_t$. Then $V_1(\mathcal{E}) = V_n(\mathcal{E}) = |\mathcal{E}| = \prod_{t\in[1..n]} c_t$. Moreover, $V_2(\mathcal{E}) = \ldots = V_{n-1}(\mathcal{E}) = \prod_{t\in[1..n]} c_t$. This is because

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

For a particular dimension t, the number of times that c_t appears in product is equal to the number of subsets $\vec{t}$ that include t. This is equal to $\binom{n-1}{q-1}$ and thus it follows that $V_q(\mathcal{E}) == \prod_{t\in[1..n]} c_t$.

7.1.2 A Bit of Information

Definition 7.3 *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 $\mathcal{E}_{\vec{t}}$ or equivalently, $|\mathcal{E}_{\vec{t}}| = |\mathcal{E}^{(0)}_{\vec{t}}| + |\mathcal{E}^{(1)}_{\vec{t}}|$.*

Intuitively, the above definition means that one can determine the bit b by knowing only the values in the q coordinates specified by $\vec{t}$. The number of different $\vec{t}$ that determine b is tied to the intuition stating that the encodings contain some number of bits "about" b. The intuition goes on to say that if the encodings contain s bits "about" b, then there must be a way of fixing b that decreases the measure of $\mathcal{E}$ by at least a factor of 2^s.

Lemma 7.4 *Let $\mathcal{E} = \mathcal{E}^{(0)} \dot\cup \mathcal{E}^{(1)}$ and let $b = 0$ on $\mathcal{E}^{(0)}$ and $b = 1$ on $\mathcal{E}^{(1)}$. Let $D \subseteq \binom{n}{q}$ be the set of projections to q coordinates that determine b and let $d = |D|$. There is a setting of $b \in \{0,1\}$ for which*

$$V_q(\mathcal{E}^{(b)}) \leq 2^{-\frac{d}{\binom{n-1}{q-1}}} \cdot V_q(\mathcal{E}).$$

Example: Consider the following case where $q = 1$, thus $\binom{n-1}{q-1} = 1$. Let

$$\mathcal{E}^{(0)} = \{0..a/2-1\}^s \times \{0..a-1\}^{n-s},$$

$$\mathcal{E}^{(1)} = \{a/2..a-1\}^s \times \{0..a-1\}^{n-s},$$

and

$$\mathcal{E} = \mathcal{E}^{(0)} \dot{\cup} \mathcal{E}^{(1)}.$$

The high order bit in each of the first s coordinates is b, and thus $d = s$. Note that

$$V_1(\mathcal{E}) = \prod_{t\in[1..n]} |\mathcal{E}_t| = \prod_{t\in[1..n]} |\{0..a-1\}| = a^n,$$

whereas

$$V_1(\mathcal{E}^{(0)}) = V_1(\mathcal{E}^{(1)}) = (a/2)^s \ \ a^{n-s},$$

and thus

$$V_1(\mathcal{E}^{(0)}) = V_1(\mathcal{E}^{(1)}) = 2^{-s} \cdot V_1(\mathcal{E}).$$

Thus, in this example for both $b = 0$ and $b = 1$ the inequality in Lemma 7.4 is an equality.

Lemma 7.4, with $D \subset \binom{n}{q}$, is used for the lower bound on the rate of a probabilistic PET system (Theorem 3.6). For the lower bound on the rate of a deterministic PET system (Theorem 3.3), we use the following corollary.

Corollary 7.5 *If every q coordinates determines the bit b, then there is a setting of $b \in \{0,1\}$ for which*

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

Proof of Corollary 7.5: In this case, $d = \binom{n}{q}$ and $d/\binom{n-1}{q-1} = n/q$. □

The intuition for this corollary is that if every q coordinates determines a bit b, then every coordinate contains $1/q$ bits "about" b or equivalently the encoding contains a total of n/q bits about b. Thus, for some setting of b, the measure on the encodings should decrease by a factor of at least $2^{n/q}$.

7.1.3 The Relationships Between the Measures

The following proof of Lemma 7.2 is a generalization of the proof found in [8]. Hölder's inequality, which is a generalization of Cauchy Schwartz's inequality, will be needed in this proof. A proof of this inequality can be found in [6].

Lemma 7.6 ([Hölder's inequality]) *For all values $v_t^{(x)}$ where $x \in [0..a-1]$ and $t \in [1..n-1]$,*

$$\sum_{x\in[0..a-1]} \left(\prod_{t\in[1..n-1]} v_t^{(x)} \right)^{\frac{1}{n-1}} \leq \prod_{t\in[1..n-1]} \left(\sum_{x\in[0..a-1]} v_t^{(x)} \right)^{\frac{1}{n-1}}.$$

The first step in the proof of Lemma 7.2 is to prove the statement for $q = n-1$.

Lemma 7.7

$$V_{n-1}(\mathcal{E}) = \left(\prod_{t\in[1..n]} \left|\mathcal{E}_{[1..n]-t}\right|\right)^{\frac{1}{n-1}} \geq \left|\mathcal{E}\right|,$$

where $|\mathcal{E}_{[1..n]-t}|$ *is the projection of* $\mathcal{E}$ *onto the dimensions*

$$\vec{t} = \langle 1, ..., t-1, t+1, ..., n\rangle \in \binom{n}{n-1}.$$

Proof of Lemma 7.7: The proof is by induction on n. The base case is $n = 2$, where the statement becomes $|\mathcal{E}_1| \cdot |\mathcal{E}_2| \geq |\mathcal{E}|$, and this is obvious.

Assume that the hypothesis is true for $n-1$. Let $\mathcal{E} \subseteq \{0..a-1\}^n$ be an arbitrary set of points in $\mathbf{Z}^n$. Partition this set of points according to the n^{th} coordinate, i.e., for $x \in [0..a-1]$, let $\mathcal{E}^{(x)}$ be the subset of $\mathcal{E}$ that has the value x in the n^{th} coordinate. The first observation to make is that for all values $x \in [0..a-1]$,

$$\left|\mathcal{E}^{(x)}\right| \;=\; \left|\mathcal{E}^{(x)}_{[1..n-1]}\right| \leq \left|\mathcal{E}_{[1..n-1]}\right|. \tag{12}$$

For each value $x \in [0..a-1]$, the set $\mathcal{E}^{(x)}_{[1..n-1]}$ is a subset of $\mathbf{Z}^{n-1}$. Therefore, the induction hypothesis can be applied giving

$$\left|\mathcal{E}^{(x)}_{[1..n-1]}\right|^{n-2} \;\leq\; \prod_{t\in[1..n-1]} \left|\mathcal{E}^{(x)}_{[1..n-1]-t}\right|. \tag{13}$$

Multiplying Equations (12) and (13) together gives

$$\left|\mathcal{E}^{(x)}\right|^{n-1} \;\leq\; \left|\mathcal{E}_{[1..n-1]}\right| \cdot \left(\prod_{t\in[1..n-1]} \left|\mathcal{E}^{(x)}_{[1..n-1]-t}\right|\right).$$

Because $\mathcal{E}^{(0)}, \ldots, \mathcal{E}^{a-1}$ is a partition of $\mathcal{E}$, it follows that

$$\left|\mathcal{E}\right| = \sum_{x\in[0..a-1]} \left|\mathcal{E}^{(x)}\right| \quad \text{and similarly} \quad \left|\mathcal{E}_{[1..n]-t}\right| = \sum_{x\in[0..a-1]} \left|\mathcal{E}^{(x)}_{[1..n-1]-t}\right|.$$

Therefore,

$$\begin{aligned}
\left|\mathcal{E}\right| &= \sum_{x\in[0..a-1]} \left|\mathcal{E}^{(x)}\right| \\
&\leq \left|\mathcal{E}_{[1..n-1]}\right|^{\frac{1}{n-1}} \cdot \sum_{x\in[0..a-1]} \left(\prod_{t\in[1..n-1]} \left|\mathcal{E}^{(x)}_{[1..n-1]-t}\right|\right)^{\frac{1}{n-1}}
\end{aligned}$$

and by Hölder's Inequality

$$\leq \left|\mathcal{E}_{[1..n-1]}\right|^{\frac{1}{n-1}} \cdot \prod_{t\in[1..n-1]} \left(\sum_{x\in[0..a-1]} \left|\mathcal{E}^{(x)}_{[1..n-1]-t}\right|\right)^{\frac{1}{n-1}}$$

$$
\begin{aligned}
&= \left|\mathcal{E}_{[1..n-1]}\right|^{\frac{1}{n-1}} \cdot \prod_{t\in[1..n-1]} \left|\mathcal{E}_{[1..n]-t}\right|^{\frac{1}{n-1}} \\
&= \prod_{t\in[1..n]} \left|\mathcal{E}_{[1..n]-t}\right|^{\frac{1}{n-1}} = V_{n-1}(\mathcal{E})
\end{aligned}
$$

This completes the induction step, proving the hypothesis for $n \geq 2$. □

Lemma 7.7 is the major step to prove Lemma 7.2.

Proof of Lemma 7.2: Clearly $V_1(\mathcal{E}) = \prod_{t\in[1..n]} |\mathcal{E}_t| \leq a^n$ and $V_n(\mathcal{E}) = |\mathcal{E}|$. Therefore, it is sufficient to prove that for all n and for all $q \leq n$,

$$V_{q-1}(\mathcal{E}) \geq V_q(\mathcal{E}).$$

For $q = n$ this has been done in Lemma 7.2. Hence fix values q, n such that $q < n$.

By definition

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

Because $\mathcal{E}_{\vec{t}}$ can be viewed as a subset of $\mathbf{Z}^q$, Lemma 7.7 implies

$$\left|\mathcal{E}_{\vec{t}}\right| \leq V_{q-1}(\mathcal{E}_{\vec{t}}) = \left(\prod_{t\in\vec{t}} \left|[\mathcal{E}_{\vec{t}}]_{\vec{t}-t}\right| \right)^{\frac{1}{q-1}} = \left(\prod_{t\in\vec{t}} \left|\mathcal{E}_{\vec{t}-t}\right| \right)^{\frac{1}{q-1}}.$$

It follows that

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

Given a particular set $\vec{u} \in \binom{n}{q-1}$ of $q-1$ dimensions, there are $n-q+1$ ways of extending it to a set $\vec{t} \in \binom{n}{q}$ of q dimensions. Therefore, for any particular $\vec{u} = \vec{t} - t$, $\left|\mathcal{E}_{\vec{t}-t}\right|$ appears in the product $n-q+1$ many times. Moreover,

$$n - q + 1 \cdot \frac{1}{(q-1)} \cdot \frac{1}{\binom{n-1}{q-1}} = \frac{1}{\binom{n-1}{q-2}}.$$

We can conclude that

$$V_q(\mathcal{E}) \leq \left(\prod_{\vec{u}\in\binom{n}{q-1}} \left|\mathcal{E}_{\vec{u}}\right| \right)^{\frac{1}{\binom{n-1}{q-2}}} = V_{q-1}(\mathcal{E}).$$

□

7.1.4 Fixing a Bit of Information Decreases the Measure $V_q(\mathcal{E})$

Before proving 7.4, the following numerical observation is needed.

Lemma 7.8 *Let $c_1, \ldots, c_d$, $c_1^{(0)}, \ldots, c_d^{(0)}$,and $c_1^{(1)}, \ldots, c_d^{(1)}$ be real positive numbers such that*

$$\forall j \in [1..d],\ c_j = c_j^{(0)} + c_j^{(1)}.$$

Define

$$c = \prod_{j\in[1..d]} c_j,\quad c^{(0)} = \prod_{j\in[1..d]} c_j^{(0)},\quad c^{(1)} = \prod_{j\in[1..d]} c_j^{(1)}.$$

There is a value of $b \in \{0,1\}$ such that

$$c^{(b)} \leq 2^{-d} \cdot c.$$

Proof of Lemma 7.8: By the arithmetic-geometric-mean-inequality

$$c^{(0)} \cdot c^{(1)} = \prod_{j\in[1..d]} c_j^{(0)} \cdot c_1^{(1)} \leq \left(\frac{1}{2}\right)^{2d} \prod_{j\in[1..d]} \left(c_j^{(0)} + c_j^{(1)}\right)^2 \leq \left(\frac{1}{2^d} c\right)^2 .$$

Hence, either $c^{(0)}$ or $c^{(1)}$ must be less than $2^{-d} \cdot c$. ❑

Proof of Lemma 7.4: Let

$$v = \prod_{\vec{t}\in D} |\mathcal{E}_{\vec{t}}|,\quad v^{(0)} = \prod_{\vec{t}\in D} \left|\mathcal{E}_{\vec{t}}^{(0)}\right|,\quad v^{(1)} = \prod_{\vec{t}\in D} \left|\mathcal{E}_{\vec{t}}^{(1)}\right|.$$

We have that

$$\forall \vec{t} \in D,\ |\mathcal{E}_{\vec{t}}| = |\mathcal{E}_{\vec{t}}^{(0)}| + |\mathcal{E}_{\vec{t}}^{(1)}|$$

The numerical observation Lemma 7.8 then gives that there is a $b \in \{0,1\}$ such that

$$v \geq 2^d \cdot v^{(b)}.$$

This gives that

$$\begin{aligned} V_q(\mathcal{E}) &= \left(\prod_{\vec{t}\in D} |\mathcal{E}_{\vec{t}}| \cdot \prod_{\vec{t}\notin D} |\mathcal{E}_{\vec{t}}|\right)^{\frac{1}{\binom{n-1}{q-1}}} \\ &\geq \left(2^d \cdot \prod_{\vec{t}\in D} \left|\mathcal{E}_{\vec{t}}^{(b)}\right| \cdot \prod_{\vec{t}\notin D} \left|\mathcal{E}_{\vec{t}}^{(b)}\right|\right)^{\frac{1}{\binom{n-1}{q-1}}} \\ &= 2^{\frac{d}{\binom{n-1}{q-1}}} \cdot V_q(\mathcal{E}^{(b)}) \end{aligned}$$

❑

7.2 The Rate Bound for a Deterministic PET System

We now have the necessary tools for proving the rate bound stated in Theorem 3.3.

Theorem 3.3 *For any PET with priority function β,* $\mathrm{rate}_\beta \leq 1$.

Proof of Theorem 3.3: As before, we view the encodings produced by a PET system geometrically, i.e., an encoding $E(M) = \langle E_1, \ldots, E_n \rangle \in \left[\{0,1\}^\ell\right]^n$ of a message M is viewed as a point in the n-dimensional lattice $\mathbf{Z}^n$, where each coordinate lies between 0 and $2^\ell - 1$. A set $\mathcal{E}$ of encodings can be viewed a set of such points.

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. Consider some message bit M_i. For $b_i \in \{0,1\}$ let $\mathcal{E}^{(b_i)} = \{E(M) \in \mathcal{E} \mid M_i = b_i\}$.

Let $q_i = \beta_i/\ell$ denote the number of packets needed to determine the message bit M_i in the PET system. Let $\vec{t} \in \binom{n}{q_i}$ denote any q_i of the n packets. Geometrically, $\mathcal{E}_{\vec{t}}$ denoted the projection of $\mathcal{E}$ onto the q_i dimensions defined by $\vec{t}$. In the PET context, it is the set of strings that the packets $\vec{t}$ can possibly contain. Any particular string in these packets determines M_i. This implies that in the geometric context, $\mathcal{E}_{\vec{t}}^{(0)} \,\dot{\cup}\, \mathcal{E}_{\vec{t}}^{(1)}$ is a partition of $\mathcal{E}_{\vec{t}}$. Hence, Corollary 7.5 gives that there is a setting of $b_i \in \{0,1\}$ for which

$$V_{q_i}(\mathcal{E}^{(b_i)}) \leq 2^{-n/q_i} \cdot V_{q_i}(\mathcal{E}) = 2^{-e/\beta_i} \cdot V_{q_i}(\mathcal{E}).$$

Using Lemma 7.2 and Corollary 7.5 alternately and since $q_1 \leq q_2 \leq \ldots \leq q_m$ we conclude that there are $b_1, \ldots, b_m \in \{0,1\}$ such that

$$\begin{array}{rccccc}
2^e & \geq & V_{q_1}(\mathcal{E}) & \geq & 2^{e/\beta_1} \cdot V_{q_1}(\mathcal{E}^{(b_1)}) \\
 & \geq & 2^{e/\beta_1} \cdot V_{q_2}(\mathcal{E}^{(b_1)}) & \geq & 2^{e/\beta_2 + e/\beta_1} \cdot V_{q_2}(\mathcal{E}^{(b_1 b_2)}) \\
 & \cdots & & & \\
 & \geq & 2^{\sum_{i \in [1..m-1]} e/\beta_i} \cdot V_{q_m}(\mathcal{E}^{(b_1 b_2 \ldots b_{m-1})}) & \geq & 2^{\sum_{i \in [1..m]} e/\beta_i} \cdot V_{q_m}(\mathcal{E}^{(b_1 b_2 \ldots b_m)}) \\
 & \geq & 2^{\sum_{i \in [1..m]} e/\beta_i} \cdot |\mathcal{E}^{(b_1 b_2 \ldots b_m)}| & &
\end{array}$$

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

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

$\square$

7.3 The Rate Bound for a Probabilistic PET System

The rate bound for probabilistic PET system is shown in a similar way. Let us recall the bound from Section 3.

Theorem 3.6 *For any probabilistic PET with priority function* β *and confidence* p, $\mathrm{rate}_\beta \leq 1/(1-p)$.

Proof of Theorem 3.6: Consider any message bit M_i and any $q_i = \beta_i/\ell$ packets $\vec{t} \in \binom{n}{q_i}$. By the definition of a probabilistic PET, the probability that the packets $\vec{t}$ determines the bit M_i is at least $(1-p)$ where the probability is over the randomness R of the system. For all R, let $D_{\langle i,R\rangle} \subseteq \binom{n}{q_i}$ be the sets of packets $\vec{t}$ that determine M_i when the random string is set to R. Let $d_{\langle i,R\rangle} = |D_{\langle i,R\rangle}|$. Let d_i be the random variable induced by R.

$$\mathrm{Exp}(d_i) = (1-p)\binom{n}{q_i}.$$

Let d be the random variable formed from the following scaled sum of the d_i.

$$d = \sum_{i\in[1..m]} \frac{d_i}{\binom{n-1}{q_i-1}}$$

$$\mathrm{Exp}(d) \;=\; (1-p)\sum_{i\in[1..m]} \frac{\binom{n}{q_i}}{\binom{n-1}{q_i-1}} \;=\; (1-p)\sum_{i\in[1..m]} \frac{e}{\beta_i} \;=\; (1-p)e\cdot \mathrm{rate}_\beta.$$

There must be some setting of the random input R that obtains this expected value. Fix such a setting. In the following, R is dropped from the notation.

Completing the proof as before

$$\begin{array}{rclcl} 2^e & \geq & V_{q_1}(\mathcal{E}) & \geq & 2^{\frac{d_1}{\binom{n-1}{q_1-1}}} \cdot V_{q_1}(\mathcal{E}^{b_1}) \\ & \cdots & & & \\ & \geq & 2^{\sum_{i\in[1..m]} \frac{d_i}{\binom{n-1}{q_i-1}}} \cdot |\mathcal{E}^{b_1 b_2 \ldots b_m}| & \geq & 2^d \end{array}$$

But by our choice of the random input R,

$$d \geq (1-p)e\cdot \mathrm{rate}_\beta$$

giving

$$\mathrm{rate}_\beta \leq \frac{1}{(1-p)}.$$

□

7.4 A Lower Bound on the Packet Size

To begin a small observation is needed.

Claim 1 *If two balls are drawn randomly without replacement from an urn containing* q *balls each colored with one of at most* c *different colors, then the probability that the two balls are the same color is at least* $1/c - 1/q$.

Proof of Claim 1: For $i \in [1..c]$, let q_i be the number of balls with the i^{th} color. The probability that the two balls are the same color is

$$\sum_{i\in[1..c]} \frac{q_i}{q}\left(\frac{q_i-1}{q}\right) = \frac{\sum_{i\in[1..c]} q_i^2}{q^2} - \frac{\sum_{i\in[1..c]} q_i}{q^2}.$$

The sum of squares is minimized when all the q_i have the same value $q_i = q/c$. The probability then is at least

$$\frac{\sum_{i\in[1..c]}(q/c)^2}{q^2} - \frac{1}{q} = \frac{1}{c} - \frac{1}{q}.$$

□

Theorem 7.9 *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 $\ell \geq \log(e/\beta) - O((e/\beta)2^{-b})$.*

The bound given in this theorem remains true even is the unique identifier of a packet (recall the convention on page 3) is not considered to be part of the packet.

Proof of Theorem 7.9: For message $M \in \{0,1\}^b$, denote the encoding by $E(M) = \langle E_1(M), \ldots E_n(M)\rangle$, where $E_t(M)$ is the contents of the t^{th} packet. Choose two messages $M_1, M_2 \in \{0,1\}^b$ randomly without replacement. There are 2^ℓ different strings that can be in the t^{th} packet. Therefore, by Claim 1, the probability that the contents of t^{th} packet is the same for the two messages is

$$\Pr_{M_1,M_2}[E_t(M_1) = E_t(M_2)] \geq 2^{-\ell} - 2^{-b}.$$

The expected number of packets that have same contents for the two messages is

$$\mathrm{Exp}_{M_1,M_2}[|\{t \mid E_t(M_1) = E_t(M_2)\}|] \geq n \cdot \left(2^{-\ell} - 2^{-b}\right) = e/\ell \cdot \left(2^{-\ell} - 2^{-b}\right).$$

There must be two distinct messages M_1 and M_2 for which the number of packets with the same contents is at least this expectation. Fix two such messages.

Any β/ℓ packets must be able to distinguish the messages M_1 and M_2. Therefore, there cannot be β/ℓ packets that have the same contents for these messages. We can conclude that

$$e/\ell \cdot \left(2^{-\ell} - 2^{-b}\right) \leq \beta/\ell$$

or

$$\ell \geq \log\left(\frac{1}{\beta/e + 2^{-b}}\right) \geq \log(e/\beta) - 1.5(e/\beta)2^{-b}.$$

□

There are two corollaries of Theorem 7.9. 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. $\beta_{\log e} \leq e^{1-\epsilon}$. Theorem 7.9 implies that a PET system with such a priority function β requires packets of size at least

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

Acknowledgment We thank Madhu Sudan for insightful remarks and discussions during the earlier stages of this work. We thank Celina Albanese for helping to clarify the presentation in this paper. We thank Oded Goldreich for allowing us to include Theorem 5.3, which originally was proved by Oded Goldreich and Michael Luby. We also thank Richard Karp for help in the proofs of Section 7.

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] E. R. Berlekamp, *Algebraic Coding Theory*, McGrawhill, New York, 1968.

[4] E. W. Biersack, *Performance evaluation of forward error correction in anATM environment*, 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] G. H. Hardy, J. E. Littlewood, G. Pólya, *Inequalities,* Cambridge University Press, 1934.

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

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

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

[10] 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.

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

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