

Packet-Based Protocol Efficiency for Wireless Communications

David Andrew Carek*
NASA Glenn Research Center, Cleveland, Ohio 44135

This paper examines the relation between bit error ratios and the effective link efficiency when transporting data with a packet-based protocol. Relations are developed to quantify the impact of a protocol's packet size and header size relative to the bit error ratio of the underlying link. These relations are examined in the context of radio transmissions that exhibit variable error conditions, such as those used in satellite, aeronautical, and other wireless networks. A comparison of two packet sizing methodologies is presented. From these relations, the true ability of a link to deliver user data, or information, is determined. Relations are developed to calculate the optimal protocol packet size for given link error characteristics. These relations could be useful in future research for developing an adaptive protocol layer. They can also be used for sizing protocols in the design of static links, where bit error ratios have small variability.

Nomenclature

B_{ER}	bit error ratio (bits errored/bits transmitted)
b_e	bits in error
b_d	bits discarded because of packet error
b_x	bits transmitted
e_h	packet header efficiency (%)
e_i	information efficiency (%)
e_p	packet delivery efficiency (%)
n	error density
S_h	packet header size (bits)
S_p	packet size (bits)
P_{ER}	packet error ratio
P_{gb}	probability of receiving a good bit
P_{gp}	probability of receiving a good packet

Numeric subscripts:

1	deterministic formulation
2	probabilistic formulation

I. Introduction

VARIOUS factors influence the true performance of wireless links in delivering information. Quite often the efficiencies of these links are specified by a bit error ratio (BER). However, this factor is not sufficient in describing the true efficiency of delivering usable user data, or information, over a communications path. The BER

Received 6 October 2004; revision received 26 January 2005; accepted for publication 21 February 2005. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States. Copies of this paper may be made for personal or internal use, on condition that the copier pay the \$10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 1542-9423/04 \$10.00 in correspondence with the CCC. This material is a work of the U.S. Government and is not subject to copyright protection in the United States.

*Professional Engineer.

only represents the efficiency of data delivered from the link layer. Additional factors come into play when using a packet-based protocol layer on top of the link layer.

The traditional layered model for communications networks, where each layer is independent of the other, can create problems in high-bit-error networks. Large inefficiencies may result if the link is designed independent of the network or transport protocol.

Packet-based protocols that perform error checking, such as TCP,¹ UDP with checksums enabled,² or Consultative Committee for Space Data Systems (CCSDS) File Delivery Protocol,³ can skew the amount of actual information that is received correctly. This is because a single bit error destroys an entire packet.

This paper develops and examines relations to correlate various protocol parameters to the link BER in order to determine the effective efficiency in delivering information across a link. Relations are also developed to calculate the optimal protocol packet size for a given BER.

The analysis considers the implications of packet sizing on network/transport layer protocols. Similar concepts have previously been presented for link layer optimization.⁴⁻⁶ This paper demonstrates that equal consideration must be given to network/transport layer packet size optimization when running packet-based protocols on top of those links. This is especially important for wireless links, such as satellite links, that do not retransmit errored packets at the link layer. The formulations developed for network/transport layer packet size optimization are similar to link layer frame optimization discussed in Ref. 6. The similarity demonstrates a potential need for layer interaction in optimizing throughput. Alternatively, the link layer protocol could be eliminated for some point-to-point applications, performing all optimizations at the network/transport layers.

While previous work mainly focuses on random error distributions, this paper examines the worst-case periodic error distribution. It is shown that assuming a periodic error distribution results in a simplified packet sizing optimization equation that works well even when errors are random.

These packet-sizing optimizations could be used to increase link availability through higher BERs or to maximize the aggregate transfer of information. They can also be used to assist in designing static links where BERs have little variability.

A. Purpose

This analysis originated from a research initiative to evaluate a highly bandwidth-asymmetric link to transmit data from the International Space Station directly to a ground station over a path that may exhibit variable BERs. The conceptual design factors included a high-rate downlink and a low-rate uplink. The primary use of the uplink was to provide an acknowledgment path at the transport layer to ensure reliable transmission of data. The large asymmetry between the uplink and downlink paths necessitated the use of very large packet sizes in order to prevent acknowledgment feedback congestion (if using TCP/IP).

The impact of large packet sizes on high BER links motivated the initial analysis in this paper. Further examination was performed to determine if packet sizes could be tuned to benefit links with variable error conditions for scenarios not constrained by particular packet sizes.

This paper examines the factors that impact the effective information efficiency of packet-based protocols. Relations are developed to determine the effective information efficiency based on a number of factors, including BER, packet size, and protocol header overhead. From these relations an optimal protocol packet size is derived.

B. Scope

While the motivation for this analysis was to develop a conceptual link design from the International Space Station, the concepts are broad enough to be applied to any packet-based communications system that exhibits variable BERs, such as low-Earth-orbit satellites, aeronautical, and other wireless communications links.

C. Definitions

The term “link” includes the radio frequency (RF) portion of the data path. It includes any hardware and electronics to handle forward error correction coding/decoding, interleaving/de-interleaving, data framing, and modulation/demodulation.

The term “protocol” generically refers to the protocol that would run in software on a processor. This could include the network and/or transport protocol layers in the layered communications model. Data managed by the protocol

is passed from the processor to the link. When there is a question of the layer of the protocol, the specific type is designated.

The term “packet” is used to refer to a defined structure of bytes sent from the protocol to the link. It includes any headers and trailers applied by the protocol and the user data carried within the packet. It does not include any additional data applied at the link layer, such as framing headers, synchronization markers, or error correction data.

The term “information” refers to the usable information, or user data, carried within a packet (i.e., the packet of data minus the headers and trailers associated with the protocol).

The term “byte” is defined as 8 bits in length, sometimes referred to as an octet in other sources.

II. Analysis

A. Overview of BER

Link quality is often specified in terms of the BER*. The BER is simply the number of bits corrupted during transmission divided by the total number of bits transmitted. The values of BER range from 0 to 1. The BER is typically expressed as an exponent, such as 10^{-6} , which indicates 1 bit error for every 10^6 bits transmitted. The smaller the BER number is (i.e., the larger the negative exponent number), the better the link quality is. For instance, a link with a BER of 10^{-8} will have fewer errors than a link with a BER of 10^{-6} .

Various types of bit errors can occur on a link.⁷ Gaussian bit errors are the most basic type, occurring as random noise. The distribution of Gaussian bit errors are random single bit errors within the data stream. Bit errors can also show up as burst errors. Burst errors cause clusters of contiguous bit errors and the clusters can be spaced randomly. Burst errors can be caused by particle interference such as rain or snow, or bursts of interfering RF signals. Another type of error is a systematic error, which can be caused by internal electronics or particular bit patterns in the transmission stream. Systematic errors can be periodic or random in distribution.

Different types of bit errors can have different impacts to a protocol’s performance for a given BER. Fig. 1 illustrates the extremes in packet delivery efficiency for different error distributions with the same BER.[†] While 90% of the transmitted bits are received correctly, there is a wide variance of usable data (from 0% to 67%). The worst-case packet loss would occur with a periodic distribution of errors. For this example it results in all of the packets being discarded even though 90% of the bits were transmitted and received correctly.

If the errors occur in bursts or a random distribution, there is a probability that more than one error will occur within a single packet. In the illustration in Fig. 1, the random distribution results in 33% of the data being usable, whereas the burst error results in 67% of the data being usable.

B. Relation of BER to Packet Size

For packet-based protocols that perform a data integrity check on a per-packet basis, such as a checksum or cyclic redundancy check (CRC), any single bit error will signify an error in the entire packet of data. A single bit error in a 1500-byte (12,000-bit) packet will result in the loss of all 1500 bytes, not just the errored bit. This is not a significant problem for links that exhibit very low BERs. However, what constitutes a low BER is very dependent on the packet size.

Table 1 shows the relation of BER to data loss for various sized packets assuming a uniform periodic[‡] distribution of errors. As seen in this table, a BER of 10^{-5} results in a 12% data loss when using a 1500-byte packet, whereas the same BER results in 100% data loss when using a 64-Kbyte[§] packet.

A BER of 10^{-6} is often considered adequate for TCP/IP; however, this is only true if the packet size is not significantly larger than 1500 bytes. As shown in Table 1, a packet size of 64 Kbytes transmitted over a 10^{-6} BER

* BER is often referred to as bit error rate. However the actual meaning is not a “rate,” it is the “ratio” of bits in error to the number of bits transmitted.

[†] The packet size and error ratio shown are for illustration purposes only and do not represent realistic values seen in practice.

[‡] A more complete efficiency calculation is developed later in this paper, which includes inefficiencies due to packet header overhead and non-uniform error distribution.

[§] The use of kilo is binary definition (i.e., $1\text{ K} = 2^{10}$ or 1024) as generally used in the computer science field. However, it should be noted that the telecommunications field often uses the SI meaning of kilo, (i.e., $1\text{ k} = 1000$), especially when referring to transmission speed bit rates.

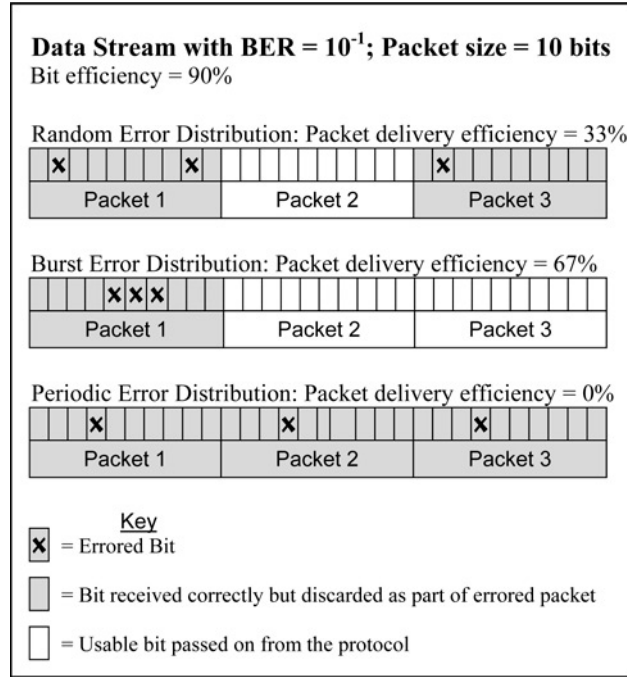


Fig. 1 Impact of error density on data efficiency.

Table 1 Percent data loss for packet size versus BER.

BER	Packet Size		
	100 byte	1500 byte	65536 byte
10^{-3}	80.000%	100.000%	100.000%
10^{-4}	8.000%	100.000%	100.000%
10^{-5}	0.800%	12.000%	100.000%
10^{-6}	0.080%	1.200%	52.429%
10^{-7}	0.008%	0.120%	5.243%
10^{-8}	0.001%	0.012%	0.524%

link will result in over 50% data loss for uniformly distributed bit errors. In other words, even though 99.9999% of the bits transmitted are received correctly, only about 50% of those bits are passed on from the protocol. This shows the substantial impact packet size can have on link efficiency. This impact may be evident on highly bandwidth asymmetric satellite links, where a larger TCP/IP packet size is required to prevent acknowledgment feedback congestion. In this scenario an alternative protocol not sensitive to acknowledgment feedback congestion may be required to increase data efficiency without increasing packet size.

C. Using Error Correction to Improve BER

Conventional protocols, such as TCP, have been optimized and enhanced over the years to support terrestrial wired links such as Ethernet. Ethernet is very different than most satellite links in that the data is delivered nearly error free at the link layer before it gets to the transport protocol layer. It is uncommon for satellite systems to have retransmission schemes at the link layer because of the added hardware complexity and cost. In addition, satellite RF and other wireless signals are much more prone to bit errors than those seen in wired links.

For communication paths where data is not easily retransmitted at the link layer, an effort is made to improve the link quality by using forward error correction (FEC). With FEC, many bit errors can be corrected using redundant

information inserted into the data stream. FEC can also provide a substantial benefit to unidirectional or broadcast links where a retransmission request for errored data is not possible.

Most satellite systems implement FEC in hardware at the link layer. Implementing FEC in a link design is a way to achieve a better BER at the expense of a lower data rate. As an example, a rate-1/2; FEC code over a 100 megabits per second (Mbps) link** would use 50 Mbps of that link for transmitting error correction information. Only 50 Mbps of the link would remain for the transmission of actual data. Reference 8 shows the theoretical improvement in BER of various FEC coding schemes for a given signal to noise ratio. A more thorough discussion of forward error correction techniques can be found in Ref. 9.

Interleaving can be used to make FEC decoding algorithms more resilient to burst errors. Interleaving is a way to disperse a contiguous burst of bit errors over a larger segment of data. By itself, interleaving could actually make packet-based protocol efficiency worse because a burst of bit errors that might only destroy one packet could be dispersed to destroy multiple packets. However, when applied to an FEC coded link, the smaller number of errors occurring in any individual code word can be corrected.

The relations developed in this paper are independent of FEC or interleaving. FEC will affect the clustering, or error density, of bit errors on a decoded signal. Reference 8 includes an example of error density for Viterbi decoded output. Interleaving can further alter the error density of the physical link. When using the relations within this paper on FEC coded and interleaved links, the BER used should be that of the link after the data passes out the de-interleaver and FEC decoder (i.e., the BER seen by the transport layer protocol on the processor).

FECs are typically selected to accommodate the worst-case link quality. The result is that a certain percentage of the available bandwidth is set aside for error correction. For systems designed with only one FEC code rate, the added error correction information is always transmitted, even when the link quality is better than the design conditions. For links that have a high variability in the BER, FEC can result in a significant reduction of information transmitted if the link quality is significantly better than the design condition for a large percentage of time. One method to optimize the quantity of data transmitted is to adaptively change the code rate. Another possibility to increase overall data quantity is to use an adaptive protocol over a link with no coding or minimal coding overhead.

D. Information Efficiency

The information efficiency describes the amount of usable information (e.g., end user data) carried over the link as a function of the total data transmitted. The total of the data transmitted includes both good packets and discarded packets (due to bit errors) with associated packet overhead for headers and trailers. The information efficiency, e_i , is the product of the packet delivery efficiency (packet loss due to bit errors), e_p , and packet header efficiency (unusable data due to packet header overhead), e_h :

$$e_i = e_p e_h \quad (1)$$

Two methodologies for determining packet delivery information efficiency are presented. The first is a deterministic approach that assumes that average bit error density per errored packet is known. The second is a probabilistic form, which is suitable for random single bit error distributions. The formulations are for the transmission of packets at the full transmission rate of a link.

1. Packet Header Efficiency

For communications links with small packet sizes, the amount of data required for headers and trailers can become a significant portion of the entire packet size. This effectively reduces the amount of useable information that can be transmitted over the link. The reduction is proportional to the packet header overhead. For instance, if a packet is 400 bytes and the header constitutes 40 bytes of that packet, then only 90% of the data within the packet is available to carry user information. The packet header efficiency, e_h , is the number of bits allocated to information (i.e., the packet size, S_p , minus the packet header size, S_h) divided by the packet size:

$$e_h = 1 - \frac{S_h}{S_p} \quad (2)$$

** Coded link speeds are usually not specified by the uncoded rate, but it is done here to show the overhead of coding.

2. Packet Delivery Efficiency

For packet-based protocols that perform a data integrity check, the usable data delivered over a communications link is a function of the number of error-free packets received or packet delivery efficiency. The packet delivery efficiency, e_p , is the ratio of usable packets to transmitted packets. This can also be expressed as one minus the packet error ratio (ratio of discarded packets to transmitted packets), P_{ER} :

$$e_p = 1 - P_{ER} \quad (3)$$

The packet error ratio can also be expressed in terms of bits (i.e., the number of bits discarded, b_d , because of packet loss, divided by the number of bits transmitted, b_x):

$$P_{ER} = \frac{b_d}{b_x} \quad (4)$$

3. Deterministic Packet Delivery Efficiency

For error distributions where no more than one error occurs within a packet, the number of bits discarded, b_d , due to bit errors is simply the number of bit errors, b_e , multiplied by the packet size, S_p (i.e., one bit error will cause one entire packet of data to be discarded). However, when error distribution is not uniform there is a possibility that more than one error may occur within one packet, thus reducing the number of bits discarded. An error density factor, n , is introduced here to account for a non-uniform error distribution:

$$b_d = \frac{b_e}{n} S_p \quad (5)$$

The error density is used to account for clustering of errors within a single packet and represents the average number of bit errors per errored packet. Clustering of errors can occur for various reasons. On FEC coded links, errors can appear as bursts of multiple bit errors. For random distributions of single bit errors there is a probability that two or more errors may occur within a single packet depending on the BER and the packet size.

By definition, the bit error ratio, B_{ER} , is

$$B_{ER} = \frac{b_e}{b_x} \quad (6)$$

Substituting b_d from Eq. (5) into Eq. (4) then solving Eq. (6) for b_x and substituting into Eq. (4) yields an expression for packet error ratio relative to the BER:

$$P_{ER} = \frac{B_{ER} S_p}{n} \quad (7)$$

Substituting Eq. (7) into Eq. (3) yields the deterministic form of packet delivery efficiency, e_{p1} , relative to the BER:

$$e_{p1} = 1 - \frac{B_{ER} S_p}{n} \quad (8)$$

Substituting e_{p1} from Eq. (8) and e_h from Eq. (2) into Eq. (1) yields a complete expression for the deterministic representation of information efficiency, e_{i1} :

$$e_{i1} = \left(1 - \frac{B_{ER} S_p}{n}\right) \left(1 - \frac{S_h}{S_p}\right) \quad (9)$$

Figure 2 shows the information efficiency from Eq. (9) expressed as a function of packet size. These curves are drawn for a standard TCP/IP packet with a header of $S_h = 320$ bits (40 bytes) and the worst-case error density of $n = 1$. The peak of each curve represents the optimal packet size for a given BER, header size, and error density. To the left of the peak is where header overhead dominates efficiency; to the right of the peak is where packet loss due to bit errors dominates efficiency.

CAREK

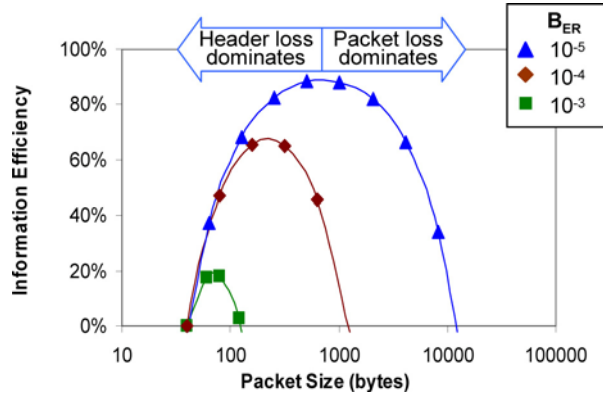


Fig. 2 Packet size vs. information efficiency (B_{ER} = variable, $S_h = 40$ bytes; $n = 1$).

Taking the first derivative of information efficiency in Eq. (9) with respect to packet size determines the slope of the curve. Since the maximum occurs where the slope is zero, setting this first derivative to zero and solving for packet size yields an equation to determine the packet size that optimizes information efficiency:

$$S_{p1} = \sqrt{\frac{S_h n}{B_{ER}}} \quad (10)$$

for $S_p > S_h$

Figure 3 shows the relation of a link's BER versus the information efficiency for various packet sizes using Eq. (9). These curves are drawn for a standard TCP/IP packet with a header of $S_h = 320$ bits (40 bytes) and the worst-case error density of $n = 1$.

The level portion of the curves represents the information efficiency where it is limited predominantly by protocol header overhead. This is very near the maximum attainable efficiency regardless of the BER for a given packet and header size. As stated earlier, what is considered an error-free link is dependent on the packet size of the protocol. The level portion of the curves represents the range of what can be considered an "error-free" link for a particular packet size and header size. Packet loss due to bit errors begins to dominate the efficiency where the curves tail downward.

Obviously, the larger packet size (64 Kbytes) has smaller header overhead for a fixed header size and therefore can achieve a higher maximum efficiency. However, as link degradation occurs, packet loss becomes the dominant

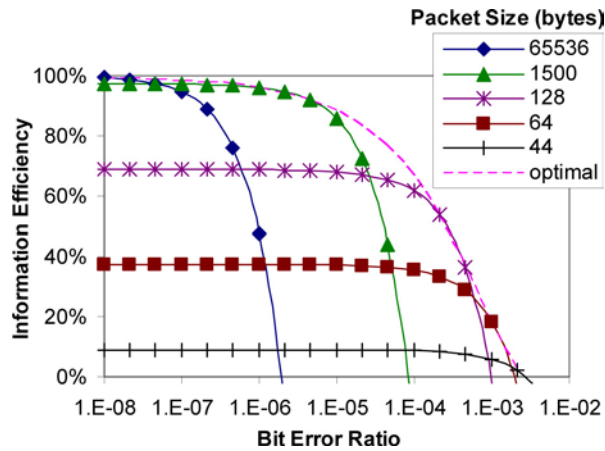


Fig. 3 BER vs. information efficiency (S_p = variable, $S_h = 40$ bytes; $n = 1$).

factor, and the larger packet size becomes a liability. This efficiency continues to decrease to zero (the point where a single bit error occurs in every packet, and therefore every packet is discarded).

Figure 3 shows that reducing the size of the packet can increase the availability of a link (i.e., the packet-based protocol can tolerate more errors without collapse). In addition, the smaller size packets run at greater efficiency for BERs greater than the crossover point of any two curves. As shown in the figure, a packet size of 1500 bytes will not be able to deliver any information at a BER of 10^{-4} , whereas a packet size of 64 bytes can still run at 35% information efficiency. As an example, a source application sending information to a transport protocol with a 1500-byte packet over a 100 Mbps link would result in the destination application not receiving any usable data. If the packet size were reduced to 64 bytes, then destination application would receive 35 Mbps of usable information for the exact same link error conditions.

The dashed curve in Fig. 3 shows the maximum efficiency that could be attained if the optimal packet size were selected for a given BER. Packet size is reduced while traversing this curve to the right (higher BERs). Reducing packet size allows continued operation at higher BERs. This increased operational range occurs until the header takes up the entire size of the packet, at which point the information efficiency is zero.

The curves in Fig. 4 represent the impact of header size for a small packet size of $S_p = 64$ bytes with the worst-case error density of $n = 1$.

When using smaller packet sizes, the header overhead can quickly become the dominant factor in link efficiency. Using a smaller packet header, higher efficiencies can be obtained. Smaller packet header definitions, such as the CCSDS Source Packet header,¹⁰ or TCP/IP compressed headers¹¹ can significantly reduce the header overhead of the protocol. As shown in Fig. 4, varying the header size for a given packet size does not change the range of link availability, but it does affect the maximum attainable efficiency as well as the operating efficiency where packet loss dominates. However, smaller headers can indirectly increase the range of link availability by allowing for smaller packet sizes as discussed below.

As previously shown in Fig. 3, you can increase link availability for increased BERs by reducing the packet size. However, the packet size cannot be reduced to a value smaller than the header size. The only way to further extend link availability would be to reduce the header size to allow even smaller packets. Figure 5 shows the extended operational range that can result by reducing header size to allow for reduced packet sizes. Figure 5 is similar to Fig. 3 except that a 6-byte compressed header is used instead of the 40-byte header. In the Fig. it is shown that the link can sustain 8% information efficiency even with a BER as high as 10^{-2} .

Figure 6 shows the impact of error density on link efficiency. As shown in the figure, a higher error density has a significant impact on improving link efficiency when operating in the region where packet loss dominates efficiency (downward curved portion). A link with a larger error density may have the same efficiency as a link with a lower BER and a smaller error density. For instance, a link with a $B_{ER} = 10^{-3}$ and an error density of $n = 100$ yields the same information efficiency as a link with a $B_{ER} = 10^{-5}$ and an error density of $n = 1$.

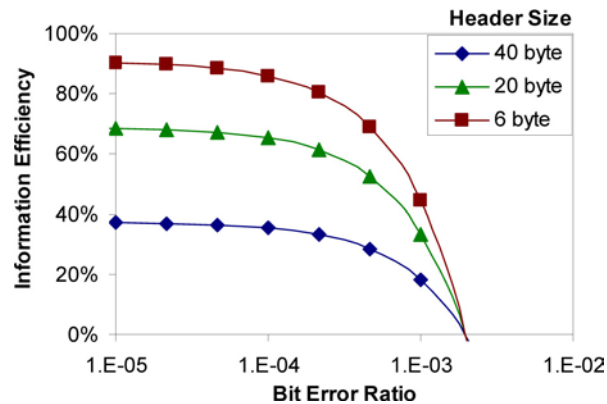


Fig. 4 BER vs. information efficiency ($S_p = 64$ bytes, $S_h =$ variable; $n = 1$).

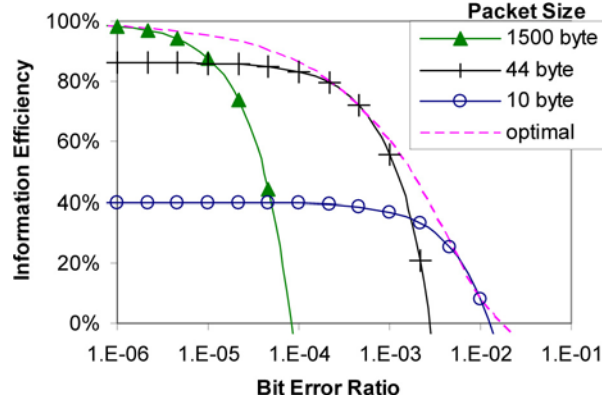


Fig. 5 BER vs. information efficiency ($S_p = \text{various}$, $S_h = 6$ bytes; $n = 1$).

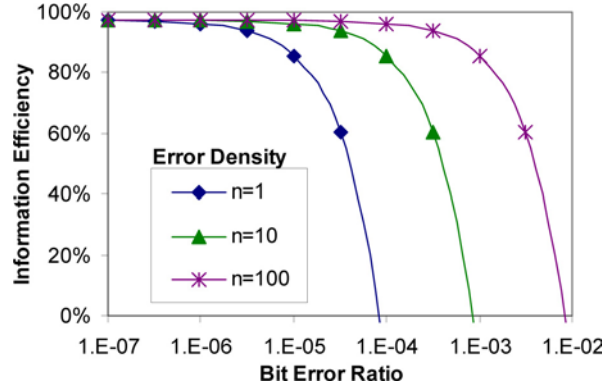


Fig. 6 BER vs. information efficiency ($S_p = 1500$ bytes, $S_h = 40$; $n = \text{variable}$).

4. Probabilistic Packet Delivery Efficiency

For links with random error patterns, the BER is a measure of probability (i.e., the probability that a bit error will occur in a stream of data). Therefore, the probability of receiving a good bit, P_{gb} , is one minus the probability of receiving an errored bit:

$$P_{gb} = 1 - B_{ER} \quad (11)$$

Using a binomial distribution to represent a random error distribution, the probability of receiving a packet of consecutive bits without error is simply the probability of receiving a good bit raised to the power of the packet size (in bits). The probability of receiving a good packet, P_{gp} , is

$$P_{gp} = (1 - B_{ER})^{S_p} \quad (12)$$

The probability of receiving a good packet is the number of good packets received divided by the total number of packets transmitted on average, which is the definition of packet delivery efficiency, e_p . Since the packet delivery efficiency, e_p , equals the probability of receiving a good packet, P_{gp} , the probabilistic form of the packet delivery efficiency, e_{p2} , is

$$e_{p2} = (1 - B_{ER})^{S_p} \quad (13)$$

Similarly, substituting the packet delivery efficiency from Eq. (13) and the packet header efficiency from Eq. (2) into Eq. (1) yields a complete expression for the probabilistic representation of information efficiency, e_{i2} :

$$e_{i2} = (1 - B_{ER})^{S_p} \left(1 - \frac{S_h}{S_p}\right) \quad (14)$$

As done previously for the deterministic form, taking the first derivative of information efficiency in Eq. (14) with respect to packet size determines the slope of the curve. Setting this to zero and solving for packet size yields an equation to determine optimal packet size:

$$S_p = \frac{S_h}{2} \left(1 + \sqrt{1 - \frac{4}{S_h \ln(1 - B_{ER})}}\right) \quad (15)$$

for $S_p > S_h$

While this equation is slightly more complicated than Eq. (10), it allows a solution without having to determine the error density. This solution is best suited for random distributions of single bit errors and may not be suitable for FEC coded links where errors occur in bursts.

E. Comparison of Deterministic and Probabilistic Efficiency and Packet Size Solutions

Figure 7 shows a comparison of Eqs. (9) and (14) for packet size versus information efficiency for several different BERs. The solid lines represent the efficiency for the deterministic Eq. (9) using the worst-case error density of $n = 1$. The dashed lines represent the efficiency for the probabilistic Eq. (14). A header size of 40 bytes is used in both equations.

The curves show that Eq. (9) and Eq. (14) are both suitable for selecting the zero slope peaks for random error distributions with BERs less than 10^{-4} for this particular header size. Where the peaks diverge (e.g., a BER of 10^{-3}), the deterministic packet sizing Eq. (10) may still predict a reasonable packet size. However, the deterministic information efficiency Eq. (9) would yield poorer results in predicting actual efficiency.

Figure 8 more closely examines what would happen if the deterministic packet sizing equation for the worst-case periodic error distribution (Eq. (10) with $n = 1$) were used for a random error distribution at a BER of 10^{-3} . The deterministic Eq. (10) calculates an optimal packet size of 71 bytes. However, the actual operating efficiency would occur at 71 bytes on the probabilistic curve (i.e. 25%). The probabilistic Eq. (15) calculates an optimal packet size of 93 bytes with a maximum information efficiency of 27% (from Eq. (14)). While Eq. (10) calculates a packet size that operates only 2% below the maximum efficiency, Eq. (9) predicts an efficiency that is 6% below the actual information efficiency for a 71 byte packet and 12% below the maximum efficiency for a 93 byte packet.

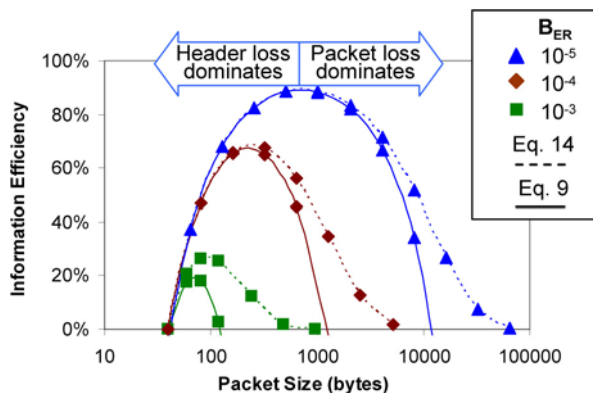


Fig. 7 Comparison of packet size versus information efficiency for periodic errors (solid line) and random errors (dashed) for various BERs ($S_p = x$ axis, $S_h = 40$ bytes; $n = 1$).

CAREK

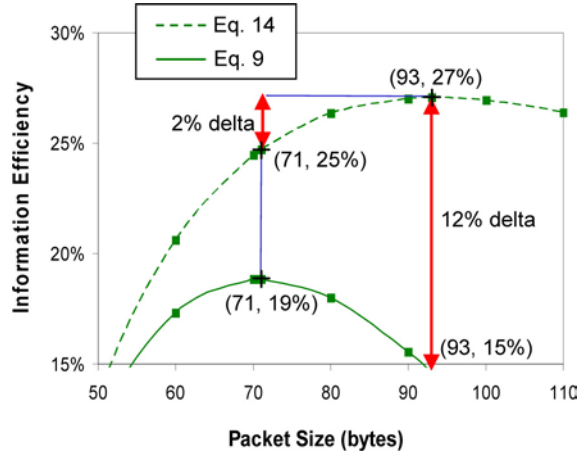


Fig. 8 Comparison probabilistic and deterministic efficiency equations ($S_p = x$ axis, $S_h = 40$ bytes; $n = 1$; $B_{ER} = 10^{-3}$).

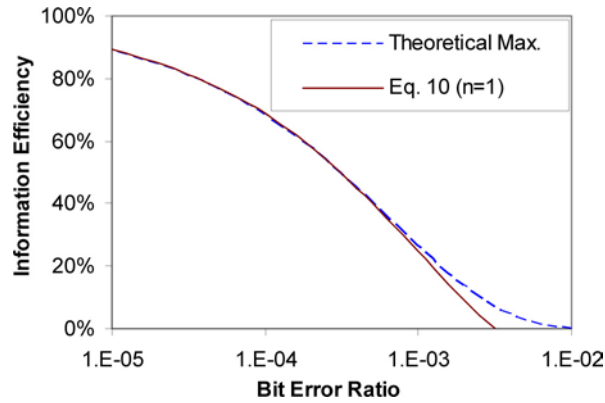


Fig. 9 BER vs. information efficiency for a random error distribution ($S_p =$ optimal based on Eq. (10), $S_h = 40$ bytes; $n = 1$).

Figure 9 shows the theoretical operating efficiency over a range of BER's when using Eq. (10) in a random error distribution. The dashed line represents the maximum attainable information efficiency. This is simply the probabilistic information efficiency from Eq. (14) calculated with the optimal probabilistic packet size from Eq. (15) at each BER point. The solid line shows the probabilistic information efficiency from Eq. (14) when the optimal packet size is calculated with Eq. (10) using the worst-case error density of $n = 1$. Equation (10) with $n = 1$ comes within 0.1 percentage points of the maximum attainable efficiency for random errors with BER's less than 10^{-4} .

By adjusting the error density, Eq. (10) yields better results for BERs greater than 10^{-4} . As shown in Fig. 10, using $n = 2$ provides results close to the theoretical maximum for any BER.

Information efficiencies for the solid line in Fig. 10 are within 1.7 percentage points of the maximum attainable information efficiency (the dashed line) across all BERs. The better performance of Eq. (10) with $n = 2$ for BERs between 10^{-2} and 10^{-3} comes at the expense of a very slight loss in efficiency for BER's in the range of 10^{-4} . This is shown better in Fig. 11, which shows the difference between the theoretical maximum and the efficiency from the use of Eq. (10).

Several different error densities are shown in Fig. 11. The worst-case error density of $n = 1$ has minimal loss for BER's less than 10^{-4} but reduces the range of link availability for high BERs. An error density of $n = 3$ provides a greater range of link availability at the expense of a 3 point percentage loss in efficiency for BER's in the range

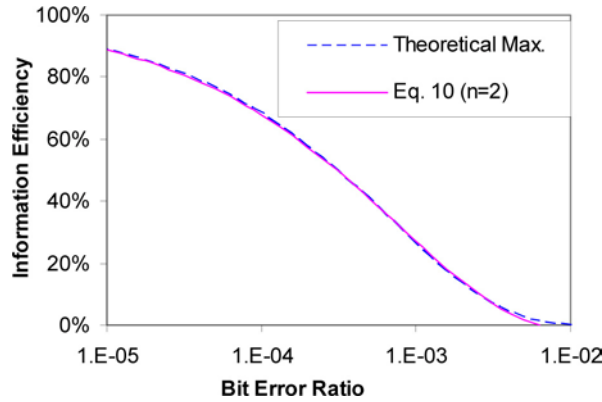


Fig. 10 BER vs. information efficiency for a random error distribution ($S_p =$ variable based on Eq. (10), $S_h = 40$ bytes; $n = 2$).

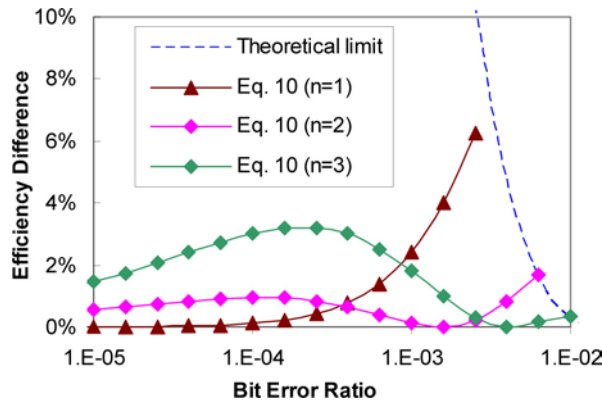


Fig. 11 BER vs. information efficiency below the maximum for a random error distribution ($S_p =$ variable based on Eq. (10), $S_h = 40$ bytes; $n =$ see legend).

of 10^{-4} . Using Eq. (10) with $n = 3$ yields an information efficiency of 1.5% which is only 0.1% less than the theoretical maximum at a BER of 6.3×10^{-3} . Using $n = 2$ provides a reasonable balance between availability and efficiency loss.

The relationship between the theoretical maximum efficiency for random error distributions and the efficiency when using a packet sized with Eq. (10) is consistent for different header sizes. As the header size is decreased, both curves shift to the right, but the proportions stay the same.

For a random error distribution, the deterministic packet sizing Eq. (10) with $n = 2$, works well over the entire range of BERs and header sizes. However the deterministic efficiency equation (Eq. (9)) would do a poor job predicting the actual efficiency for random error distributions.

III. Discussion

The preceding sections describe two different ways to determine optimal packet size for packet-based protocols. The first form is called a deterministic approach, which uses an error density factor to account for non-uniform error distribution. The second form is called a probabilistic approach, which assumes a random distribution of single bit errors. Each solution has its own problems and limitations. In the deterministic form, the error density used to account for non-uniform error distributions would be difficult to determine reliably for anything other than the random error distribution. And if using a random distribution, a lookup table would be required to correlate packet size with the density factor, turning the solution into an iterative procedure.

The error density, n , in Eqs. (9) and (10) is used to account for both the probability that more than one error will occur within a packet and the clustering of errors into bursts. For random error distributions, n can be set to the average number of errors per packet using a random probability distribution for a particular packet size. However, using $n = 2$ provides reasonable results for all BERs in a random error distribution. The n factor can also be used to account for burst type errors if the average number of consecutive bit errors can be determined. Setting $n = 1$ is equivalent to a periodic distribution of single bit errors. This is the worst-case error distribution, since every bit error results in the loss of a complete packet of data. This periodic distribution is a possible scenario that could occur because of systematic bit errors. However, these types of errors are generally designed out of the system if the error source can be determined. One can envision that the periodic distribution could also occur from external periodic RF interference at the receiver. If the error source has the potential to be periodic, then using the worst-case periodic error density ($n = 1$) would make a good design assumption.

The probabilistic representation of information efficiency and optimal packet size in Eqs. (14) and (15) eliminates the need for the probability factor, n . This allows a direct solution for the optimal packet size that maximizes information efficiency when the error distribution is known to consist of random single bit errors. The probabilistic form would yield good results for a link with no forward error correction (FEC) coding, where the errors are due to random RF noise sources. However, this solution does not account for burst-type errors that might occur on FEC coded links. Burst errors would require additional compensation. One possibility would be to divide the BER by the average burst density. However, depending on the burst size, there is a probability that the burst may span multiple packets. Using the probabilistic form for links with burst errors requires further investigation.

The results shown in this paper demonstrate that protocol packet size and header size selection can have a significant impact on effective information efficiency for links with high BERs.

These relations can be used for developing an adaptive protocol layer to increase link availability while maintaining maximum efficiency. Protocol packet size is generally an easily modified parameter. As BERs increase, a smaller packet size can be selected prior to link collapse, thereby increasing link availability. As link quality improves, the packet size can be increased to increase the effective link information efficiency.

The results in this paper can also be directly applied to link layer information efficiency by substituting the packet size with the link frame size, and the packet header size with the link header size (including header, synchronization data, and trailers). This could be useful in determining optimal frame sizes for link layer data structures when a system is designed for a given BER.

One problem in developing an adaptive protocol is accurately determining BERs. The traditional layered view of communications generally prevents the protocol layer from knowing the error conditions occurring in the link layer. While a protocol can determine that a packet has an error, it cannot determine how many errors occurred or the distribution within. If a uniform error distribution is assumed, then the BER can be determined by the transport protocol, by assuming each packet error correlates to one bit error.

It would be possible to maintain the layered concept if the link layer protocol assumed a particular overhead for the network and transport layers. The link layer could set the frame size of the link to the optimal efficiency, and the network/transport layer could set the packet size by using a process such as maximum transmission unit path discovery.¹²

However, confusion arises in network-routed paths, in that the receiver may never get the packet if the header is corrupted and dropped at a router. A mechanism would be required in routed environments to determine if the packet was dropped because of an error in the network or dropped because of a router queue overflow. Determining bit errors over the complete end-to-end path of a routed environment could be a formidable challenge.

Another difficulty in error estimation is that the device that is most capable of determining the BER, the receiver, does not need the data. The sender needs the BER data from the receiver. This necessitates an additional messaging scheme at the protocol layer to feed back the error information to the sender.

In addition to adaptive packet sizing, the formulations developed within this paper are useful for designing static links where BERs have low variability. In practice, as packet size increases, the probability of multiple errors occurring within one packet also increases; thereby increasing the error density. If the error distribution is random, then the probabilistic form of the optimal packet size solution (Eq. (15)) provides direct results, without requiring the error density factor, n . Alternatively, the deterministic packet sizing solution (Eq. (10)) can be used with an error density of $n = 2$ to provide reasonable results for random error distributions.

IV. Conclusion

Wireless link designers should consider the implications of bit error ratios on the effective link efficiency when running packet-based protocols, especially for links with high bit error ratios or very large packet sizes. Large inefficiencies may result if the link is designed independently of the protocol. This may be more significant for power-constrained systems, which may need to run at high bit error ratios.

For packet-based protocols that allow variation of packet sizes, optimizations can be made to extend link availability to higher bit error ratios and increase the effective link information efficiency.

Selecting the optimal packet size yields the maximum theoretical efficiency. This efficiency might be possible using an unreliable protocol such as UDP. However, other factors may further reduce the efficiency, such as link layer inefficiencies due to integral order mismatches between link frame sizes and packet sizes. Reliable transport protocols, such as TCP, also degrade efficiency below the theoretical maximum because congestion is assumed to cause all packet loss. This needlessly reduces the transmission rate when loss is actually due to corruption. A rate-based variation of TCP that disables congestion control, such as SCPS-TP (CCSDS 714.0-B-1: Space Communications Protocol Specification (SCPS) - Transport Protocol (SCPS-TP). Blue Book. Issue 1. May 1999.), would be required to approach maximum theoretical efficiencies for high bit error ratios.

Acknowledgments

I would like to thank Dr. Obed Sands, Mr. Mark Allman, and Mr. David Irimes, who aided in the technical review of this paper. In addition, I would like to thank Lilly Facca, Sina Javidi, and Rich Kunath, formerly of the Space Communications Office, for support and funding of this work.

Suggested Further Reading

Beyond Bit Error Ratio—Gain New Insight from Studying Error Distributions, G. M. Foster, T. Waschura, Agilent Technologies, September 26, 2002 <http://literature.agilent.com/litweb/pdf/5988-8037EN.pdf> (cited May 2005).

Influence of the Bit Error Ratio on the Frame Format in Communication Protocols Design for an Infrared Diffusive Channel, Girsha V. Spasov, Yovko D. Lambrev <http://yovko.net/pdf/irdc.pdf> (cited May 2005).

Stevens, W., "TCP/IP Illustrated, Volume 1: The Protocols", Addison-Wesley, 1994.

TCP/IP Tutorial and Technical Overview; Adolfo Rodriguez, John Gatrell, John Karas, Roland Peschke <http://www.redbooks.ibm.com/pubs/pdfs/redbooks/gg243376.pdf> (cited May 2005).

References

¹Postel, J., "Transmission Control Protocol," Internet Engineering Task Force RFC 793 (STD0007), 1981, Available online at <ftp://ftp.rfc-editor.org/in-notes/rfc793.txt> (cited May 2005).

²Postel, J., "User Datagram Protocol," Internet Engineering Task Force RFC 768 (STD0006), 1980, Available online at <ftp://ftp.rfc-editor.org/in-notes/rfc768.txt> (cited May 2005).

³CCSDS 727.0-B-1: CCSDS File Delivery Protocol (CFDP), Blue Book, Issue 1, Jan. 2002.

⁴Modiano, E., "An adaptive algorithm for optimizing the packet size used in wireless ARQ protocols," *Wireless Networks*, Vol. 5, 1999, pp. 279–286.

⁵Siew, C. K., Goodman, D. J., "Packet Data Transmission Over Mobile Radio Channels," *IEEE Transactions on Vehicular Technology*, Vol. 38, No. 2, 1989.

⁶Schwartz, M., "Telecommunication Networks Protocols, Modeling and Analysis," Addison-Wesley, 1987, pp. 131–134.

⁷An Introduction to Error Location Analysis, Are all your errors truly random?, Application Note 1550-2, Agilent Technologies, 2000, Available online at <http://literature.agilent.com/litweb/pdf/5980-0648E.pdf> (cited May 2005).

⁸CCSDS 700.0-G-3: Advanced Orbiting Systems, Networks and Data Links: Summary of Concept, Rationale and Performance, Green Book, Issue 3, November 1992, Available online at <http://www.ccsds.org/documents/700x0g3.pdf>.

⁹CCSDS 101.0-B-6: Telemetry Channel Coding, Blue Book, Issue 6, October 2002, Available online at <http://www.ccsds.org/documents/101x0b6.pdf> (cited May 2005).

¹⁰CCSDS 102.0-B-5, Packet Telemetry, Blue Book, Issue 5, Nov. 2000.

¹¹M. Degermark, B., Nordgren, S. Pink, RFC 2507, IP Header Compression, Feb. 1999.

¹²Path MTU Discovery, RFC1191, Available online at [www.rfc.net /rfc1191.html](http://www.rfc.net/rfc1191.html) (cited May 2005).