LIDS-P-2187

DESIGN OF ERROR DETECTION SCHEME FOR CLASS C SERVICE IN ATM

June, 1993

Jane M. Simmons and Robert G. Gallager
MIT, Laboratory for Information and Decision Systems

Abstract: We present a logical approach to designing an effective and efficient error detection scheme for ATM. We specifically look at providing error protection for the class C service of ATM, which is a connection oriented, variable bit rate service, with no required timing between source and destination. Our resulting scheme is similar to the scheme proposed by the CCITT in AAL 5. We propose to add a 34 bit CRC to each frame. Our proposal also includes a modification of the mechanism for preventing misdirected cells that is used at the ATM layer.

This work was funded by the NSF, DARPA, and ATT Bell Laboratories.

1. INTRODUCTION

Asynchronous Transfer Mode (ATM) is a network protocol currently being designed for use on Broadband Integrated Services Digital Network (B-ISDN) systems. ATM provides a common format for transmitting voice, data, and video over B-ISDN systems. The ATM Adaptation Layer (AAL) divides frames into fixed length segments at the transmitter. The segments are passed down to the ATM layer, which adds a fixed amount of control information to each segment to form what is called a cell. The ATM layer is responsible for cell routing. At the receiver, the AAL is responsible for reconstructing the frame and checking the integrity of the frame. (Unless otherwise indicated, we will use the term 'ATM' to refer to ATM systems and not the specific ATM layer.)

We define an undetected error event as occurring when a frame that contains any type of error is accepted by the AAL at the receiver as error-free. Error protection mechanisms need to be added at the ATM and AAL layers to ensure that the rate of undetected errors is below an acceptable threshold. In this paper, we specifically look at providing error protection for the class C service of ATM, which is a connection oriented, variable bit rate service, with no required timing between source and destination. The CCITT has proposed two error detection schemes for class C traffic, as part of the protocols referred to as AAL 3/4 [1] and AAL 5 [2]. However, it appears there is no accepted methodology for designing error detection schemes.

The purpose of this paper is to present a systematic approach to designing an effective and efficient error detection scheme for ATM. Our resulting scheme is similar to the scheme proposed in AAL 5. We propose to add a 34 bit cyclic redundancy check (CRC) to each frame; the AAL 5 proposal adds a 32 bit CRC and a length field to each frame. The AAL 3/4 proposal adds a CRC at the segment level rather than the frame level, and includes a large number of other error detection fields. Our proposal also includes a modification of the mechanism for preventing misdirected cells that is used at the ATM layer.

In section 2, we discuss the general properties of ATM networks. Only a few characteristics of ATM networks are relevant to our analysis. In section 3, we review the properties of CRCs since they play an important role in the various error detection schemes proposed for ATM. In section 4, we examine the error characteristics expected in ATM networks. We step through the design process of an error detection scheme for ATM in section 5. This is followed by a comparison with AAL 5 and a discussion of some of the shortcomings of the AAL 3/4 scheme.

AAL 3/4 is also used for class D traffic which is connectionless. In this paper, we do not specifically address error detection issues related to connectionless traffic, although much of our

analysis is applicable. The main differences are that routing cells and identifying cells that belong to a frame are handled differently for connectionless traffic, so that different mechanisms may be needed to handle routing and identification errors.

2. GENERAL PROPERTIES OF ATM NETWORKS

Frames can be variable in length, with the maximum length being 65,536 bytes.[3] Frames are broken up at the transmitter into 48 byte segments; a 5 byte header is added to each segment to form a 53 byte cell. The frame is reconstructed only at the destination. The cell header contains a virtual channel identifier/virtual path identifier (VCI/VPI) field that is used to route the cell to its destination. It is expected that all cells of a frame will follow the same path, and will arrive at the destination in the same order in which they were sent. Duplicate cells are not expected to occur.

There needs to be some method of indicating the last cell of a frame so that frames can be reconstructed correctly. It seems natural to address this issue in the AAL since this layer deals with frame reconstruction. Indeed, in the AAL 3/4 scheme, there are two bits in the segment that are used to indicate whether a cell is the first, a middle, or the last cell of the frame. As we will see, however, including these bits in the segment is not advantageous from the point of view of error detection. It is better to include an END flag in the cell header, despite the fact that the ATM layer does not make use of the flag. Thus, we will assume in the analysis below that there is a one bit END flag in the cell header to indicate the last cell of a frame.

3. REVIEW OF CYCLICAL REDUNDANCY CHECKS

Below, we review the error detection properties of CRCs. For a more complete discussion, refer to [4]. Throughout the paper, we will use the term 'CRC' to refer to an Extended Hamming Code CRC.

Assume we are using a CRC of length L to check the integrity of a data block of length K. Then L should be chosen such that [5]:

$$2^{L-1}-L \geq K+1 \qquad (1)$$

Such a CRC is guaranteed to detect all single, double, and triple bit errors that occur in the string of bits comprising the data and the CRC. Thus, at least four bit errors must occur in order for the CRC to fail to detect an error. If an error occurs such that the string of bits is random, then an L bit CRC fails to detect the error with probability 2^{-L}.[5]

The strings of bits comprising the data and the CRC are actually codewords in a code with a minimum distance of four. (If (1) is satisfied with strict inequality, the minimum distance can

be greater than four [4].) Not all patterns of four bit errors will cause the bit errors to go undetected; only those patterns that are themselves codewords will cause undetected errors. Bit errors in any given three locations uniquely determines where the fourth bit error must occur to possibly cause the CRC to fail to detect the error. If this were not true then there would be codewords that differ in only two positions, which contradicts the fact that the minimum distance is four. Thus, letting T represent the total number of data and CRC bits, the number of possible four bit patterns that can cause an undetected error can be upper bounded by:

$$\binom{T}{3}\frac{1}{4} \qquad (2)$$

The factor of 1/4 is necessary since there are 4 ways of choosing 3 bit positions out of each four-bit error pattern.

The CRC also can be used to correct single bit errors. However, if a CRC is used to correct errors, it increases the probability an error will not be detected. Three bit errors may appear to be a single bit error, and the CRC will 'correct' the error to the wrong value. Thus, three bit errors rather than four can result in an undetected error. Also, if a burst error hits such that the data and CRC bits are random, then a CRC in the correction mode will not detect the error if the resulting bit pattern matches a codeword or differs from a codeword in one bit position. There are 2^K possible codewords (corresponding to each possible data string); there are (K+L) sequences that are at distance one away from a given codeword. After the burst error hits, any given sequence will occur with probability $2^{-(K+L)}$. Thus, the probability the CRC will not detect the burst error is:

$$\frac{2^K + (K+L)2^K}{2^{K+L}} = (K + L + 1)\, 2^{-L} \qquad (3)$$

4. ERROR CHARACTERISTICS OF ATM NETWORKS

In this section we examine the underlying error characteristics of the ATM network. We assume it will be run over fiber optic lines. The three factors of concern are random bit errors, burst errors, and congestion, each of which is discussed below. Let P_R denote the probability of random bit errors, P_B the expected fraction of cells affected by burst errors, and P_C the expected fraction of cells dropped due to congestion.

Random Bit Errors: We assume independent random bit errors occur on a fiber optic line with probability 10^{-8}. This is probably an overestimate of such bit errors by several orders of magnitude. However, our calculations show that even with this conservative estimate, random bit errors are not expected to be the dominant cause of most error scenarios in ATM systems.

Even if the bit error events were correlated rather than independent, our results would not significantly change.

Burst Errors: In one study of a fiber optic system, it was found that the chief cause of burst errors is protection switching.[6] This occurs when a failed repeater causes the data to be switched from the original line to a protection line. The study showed that the mean time between these events is approximately four days, and each event results in error bursts of duration 20 to 40 msec, during which the bit error rate is .5. Assuming an average burst length of 30 msec., the fraction of time spent in such bursts is $9x10^{-8}$. At a data rate of 150 Mb/sec, about 10600 cells will be affected by a 30 msec. burst. It is unlikely that this is the only type of burst error we need to consider. In calculating the probability of various error scenarios, we will use $9x10^{-8}$ as the probability of a cell being hit by a burst error, but we will ensure robustness by making reasonable worst case assumptions as to which bits of the cell are actually affected by the burst.

Congestion: It is very difficult to estimate statistics on the expected congestion in ATM systems, due to the highly variable nature of the traffic in the network. However, in general, the design objective is to limit the end-to-end cell loss rate to 10^{-6}.[6] Therefore, we will use 10^{-6} as the probability a cell is dropped due to congestion.

5. DESIGN OF ERROR PROTECTION SCHEME

The underlying errors discussed in the previous section lead to the following error scenarios: misdirected cells, lost cells, bit errors in the data, and errors in the END flag. Our unit of measure for evaluating an error detection scheme is the expected number of frames, out of those transmitted on a line per year, for which errors are not detected by the combination of the ATM and AAL layers. We must design the error protection scheme such that the expected frequency of undetected errors is below some acceptable threshold. A realistic goal is to reduce the expected frequency of undetected errors on any given data line to no more than one per year. To provide some margin in achieving this goal, and to ensure a low rate of error even in the case of multiple lines feeding into a receiver, we use 10^{-3} as our desired maximum expected annual undetected error frequency per line. Obviously, this goal is somewhat arbitrary. The main point is we want undetected errors to be a very infrequent event.

Below we present a logical approach to designing an error protection scheme that effectively and efficiently deals with the various error scenarios. Throughout the analysis, we assume the data rate of the links is 150 Mb/sec; thus, about 10^{13} cells can travel over a data line per year. For simplicity, we will generally assume that frames are comprised of N cells (or

segments); thus, the number of frames sent per year, per line is $\frac{10^{13}}{N}$. We assume $P_R = 10^{-8}$, $P_B = 9x10^{-8}$, and $P_C = 10^{-6}$.

5.1 Prevention of Misdirected Cells

Of all the various error scenarios, misdirected cells can be considered to be the most serious. Misdirected data is a potential security threat whether or not it is detected. Thus, preventing data from being sent to the wrong destination is important, as opposed to just detecting the stray data after it's reached the incorrect destination. Thus, the first step of the design process is to provide enough protection to reduce the frequency of misdirected cells to a level that network users will find satisfactory.

Misdirection occurs when an undetected error occurs in the VCI/VPI field of the cell header, and the 'new' VCI/VPI matches an entry in a node's routing table. Misdirection can be prevented by detecting errors that occur in the VCI/VPI field. Only the cell header is examined at the intermediate nodes; thus, any error prevention mechanism must be included per cell, in the cell header. The CCITT has specified that the cell header include a CRC that checks on the contents of the header, which is a reasonable decision. The number of bits in the cell header, excluding the CRC, is 32. From equation (1), we know that the length of the CRC should be at least 7 bits. The CCITT has chosen the length to be 8 bits, so that it results in a cell header of size exactly 5 bytes.

The CCITT has chosen to use the header CRC in a two-state correction/detection mode.[7] The default state is that the node uses the CRC to correct any single bit error, so that fewer cells will be dropped. The drawback is that three or more bit errors may appear to be a single bit error, in which case the cell header is 'corrected' to the wrong value. The possibility of this occurring is greatest when a burst error has occurred. To counteract this, once the node detects that a cell has an error in its header, it goes into a detection-only state. It returns to the correction state only after it has received a cell that it perceives as having an error-free header. The state diagram is shown in Figure 1.

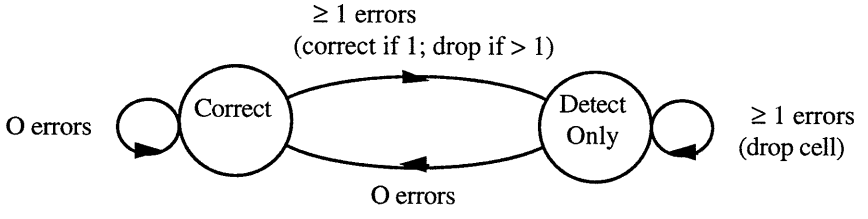

Figure 1 State diagram for 2 state correction/detection option for CRC.

We propose that the 8 bit cell header CRC be used in a four-state mode. This works similarly to the two-state option, except that after two or more errors have been detected in a cell header, or after errors have been detected in two consecutive cell headers, the intermediate node will drop all cells until it receives <u>two</u> consecutive cells that it perceives as having error-free headers. Thus, as shown in Figure 2, the transition from state 3 to state 4 results in a dropped cell even though the cell is perceived to be error-free. The rationale for this is that during a burst error, it forces the CRC of two consecutive cells to fail before a cell is accepted. We are willing to lose an additional cell during a burst error in order to gain greater protection against misdirection. As will be shown at the end of this section, the resulting increase in the number of dropped cells should be insignificant.

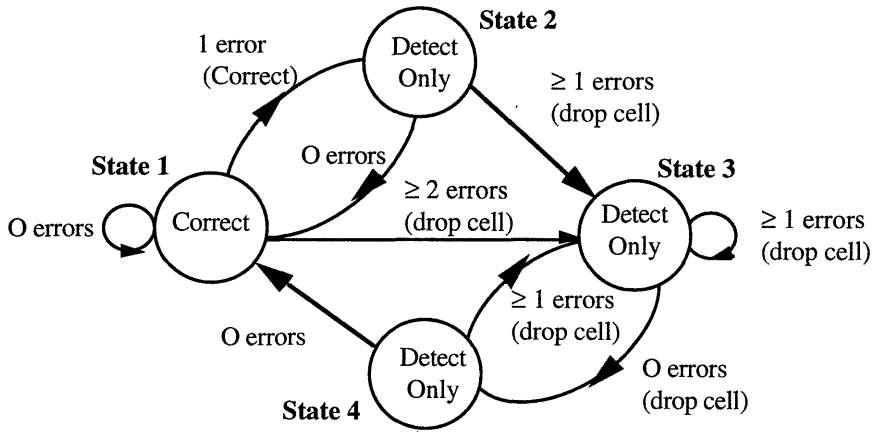

Figure 2 State diagram for 4 state correction/detection option for CRC.

Below, we examine the probability of misdirected cells due to burst errors and random bit errors, under these two header CRC schemes. We will make the worst case assumption that any undetected errors in the VCI/VPI field will cause the cell to be misdirected.

First consider burst errors. From section 4, we assume P_B is 9×10^{-8}, and we expect about 10600 cells to be hit by the average error burst. We assume the entire cell header has been hit by the burst, resulting in a completely random string. With the two-state CRC option, when the first cell affected by the error burst arrives at a node, the node is likely to be in the correcting state. Thus, using equation (3), the first cell in the error burst will be misdirected with probability $41 \cdot 2^{-8}$. After the first cell in the error burst, the node is likely to be in the detect-only state, so the other cells affected by the burst will be misdirected with probability 2^{-8}. Thus, the probability of any given cell being misdirected is about:

$$P_B\left(\frac{41\cdot 2^{-8}}{10600} + 2^{-8}\right) \approx P_B\, 2^{-8} = 3\text{x}10^{-10}.$$

With the four-state option, the first cell affected by the burst is misdirected with probability about $41\cdot 2^{-8}$. After the first cell, the node will be in state 1 with probability 2^{-8}, and in state 2 with probability $40\cdot 2^{-8}$. Thus, the second cell in the burst is misdirected with probability $((2^{-8})(41\cdot 2^{-8}) + 40\cdot 2^{-8}\,(2^{-8}))$. The remaining cells affected by the burst are misdirected with probability about 2^{-16}. Thus, the probability of any given cell being misdirected is about:

$$P_B\left(\frac{41\cdot 2^{-8}}{10600} + \frac{81\cdot 2^{-16}}{10600} + 2^{-16}\right) \approx P_B\, 2^{-15} = 2\text{x}10^{-12}.$$

Next consider random bit errors. With either the two-state or four-state option, when the CRC is in the correction state, three bit errors can result in a misdirected cell. The probability of this event can be approximated by: $\binom{40}{3} P_R^3 \approx 1\text{x}10^{-20}$. This is negligible compared to the misdirection probabilities due to burst errors.

We conclude that burst errors are the chief cause of misdirected cells, and that the four-state CRC option is more effective in dealing with this error event. The four-state option provides us with a factor of 2^{-7} benefit while incurring only a small penalty in term of complexity.

The four-state option results in a higher rate of cell loss than the two-state method, but the difference is insignificant. With the two-state option, a cell will be dropped if there are two or more bit errors in its header, or if there is at least one bit error in both its header and the previous cell's header. The probability of this occurring due to random bit errors is on the order of: $\left(\binom{80}{2} - \binom{40}{2}\right) P_R^2 \approx 2\text{x}10^{-13}$. With the four-state method, a cell will also be dropped if there are two or more bit errors in the previous cell's header. The overall probability of dropped cells due to random bit errors is then:

$$\binom{80}{2} P_R^2 \approx 3\text{x}10^{-13} \quad (4)$$

When burst errors occur, the four-state method will result in one extra cell being dropped. If thousands of cells are lost due to a burst error, then dropping one extra cell will not significantly increase the cell drop rate. In the worst case, where all burst errors are short enough that only one cell is affected, then the cell loss rate due to burst errors doubles. However, we expect that the dominant cause of lost cells will still be congestion.

We will assume for the remainder of our analysis that an 8 bit CRC is present in the cell header, and operates in a four-state correction/detection mode. Thus, the probability of a cell being misdirected is $P_B 2^{-15} = 2\text{x}10^{-12}$. At a data rate of 150 Mb/sec, about 10^{13} cells can travel over a data line per year. Thus, roughly 20 misdirected cells are expected per year per line.

5.2 Detection of Bit Errors in Data

Of all the error scenarios, the one with the least number of options for error detection is that of bit errors in the data. In this error scenario, the frame is intact except for errors in the data; the only method of detecting the error is to include some sort of redundancy check on the data. From the point of view of efficiency, it makes sense to deal with this error scenario next since the mechanism chosen to detect this error may also help detect other error scenarios, but the converse is less likely to be true.

We assume a CRC will be used to detect errored data. There are two options: a CRC per segment, which checks on the contents of the segment, or a per-frame CRC, which checks on the frame as a whole. As shown below, the per-frame CRC is the more powerful option.

Both burst errors and random bit errors can cause errors in the data. In section 4, we stated that burst errors in ATM are expected to be long; the average burst error is expected to affect about 10600 consecutive cells. However, short error bursts that only affect the data portion of a cell are more difficult to detect since they are likely to produce fewer inconsistencies. Since we can not be sure exactly what type of burst errors to expect, and since robustness is very important in designing an error detection scheme, we will make the worst case assumption that burst errors are short (i.e., shorter than the length of a cell) and do not affect any of the control information in the cell, such as the VCI/VPI field. With these assumptions, the probability a frame of N cells will contain data that has been corrupted by a burst error is about NP_B. The probability a frame of N cells contains a random bit error is about $(N)(48)(8)P_R$.

Note that the unit of retransmission in ATM is a frame, rather than an individual segment.

5.2.1 Per-Segment CRC

The size of a segment is 384 bits. Thus, the size of the per-segment CRC must be at least 10 bits in order to satisfy the inequality in (1). Assuming a 10 bit per-segment CRC is implemented as part of the 384 bits of the segment, then using equation (2) we can upper bound the probability of four random bit errors occurring in a segment and going undetected by: $\binom{384}{3} \frac{1}{4} P_R^4$. (More than four bit errors can also result in an undetected error, but P_R is so small that these events can be ignored.) A frame will contain an undetected bit error if any of the segments contain an undetected bit error. Thus, if frames are comprised of N cells, the expected annual frequency of frames with undetected random bit errors in the data, per line, is: $\frac{10^{13}}{N} N \binom{384}{3} \frac{1}{4} P_R^4 \approx 2 \times 10^{-13}$. Thus, the per-segment CRC provides very good protection against random bit errors.

The 10 bit CRC will fail to detect a burst error in the segment with probability 2^{-10}. With the assumption that cells are hit independently by short burst errors with probability P_B, the expected annual frequency of frames with undetected bit errors in the data due to burst errors, per line, is about: $\frac{10^{13}}{N} N P_B \, 2^{-10} = 9 \times 10^2$. Even if just 10% of the bursts are short, the expected number of undetected errors in the data due to burst errors is 90. This is not below the 10^{-3} threshold. In order to meet our goal, we would need to increase the length of the CRC to 30 bits. However, this would be a lot of overhead to add to each segment. One reason for this seemingly poor performance of the per-segment CRC is that we are looking at the worst case burst error scenario. However, as we show below, even with this worst case assumption, a per-frame CRC can meet the goal of 10^{-3} without a lot of overhead.

5.2.2 Per-Frame CRC

The maximum size of a frame is 2^{19} bits. Thus, the size of the per-frame CRC must be at least 21 bits in order to satisfy the inequality in (1). Assuming a 21 bit per-frame CRC is implemented, and assuming frames consist of N cells, then, using equation (2), we can upper bound the probability of four random bit errors occurring in a frame and going undetected by: $\binom{N\,384}{3} \frac{1}{4} P_R^4$. This results in an expected annual frequency of frames with undetected random bit errors in the data, per line, of about: $\frac{10^{13}}{N} N^3 \, 384^3 \, \frac{1}{24} P_R^4$. Letting N equal 1366, which is the maximum number of cells per frame, this frequency equals 4×10^{-7}, which easily satisfies the goal of 10^{-3}.

The 21 bit per-frame CRC fails to detect a burst error that hits the frame with probability 2^{-21}. Thus, with the worst case assumption that burst errors are short, the expected annual frequency of frames with undetected bit errors in the data due to burst errors, per line, is about: $\frac{10^{13}}{N} N \, P_B \, 2^{-21} = .4$. This does not meet our goal of 10^{-3}. However, if we increase the length of the per-frame CRC to 30 bits, we can meet our goal. 30 bits per frame is still not a lot of overhead; thus, this option is feasible.

We conclude that a per-frame CRC of length at least 30 bits should be included as part of the error detection scheme. This CRC is in addition to the 8 bit CRC in each cell header.

5.3 Detection of Lost Cells

Recall that we assume there is a one-bit flag in the cell header that indicates whether a cell is the last cell in the frame. This leads to two different lost cell scenarios. First, we look at the case where a non-END cell is lost, so that the frame has too few cells. Secondly, we look at the

case where an END cell is lost, so that the cells of one frame are merged with the cells of a subsequent frame.

5.3.1 Non-END Cell Lost

Congestion, burst errors, and random bit errors all can cause lost cells, but congestion is the dominant cause. It is likely that congestion will occur in a burst and will result in entire frames being dropped; it is not likely to affect the non-END cells of a frame without affecting the END cell. Nevertheless, we use the union bound, which shows that the probability of losing at least one non-END cell from a frame of N cells is at most $(N-1)P_C = (N-1)10^{-6}$.

As with congestion, we expect burst errors to affect entire frames. However, if we make the worst case assumption that burst errors are shorter than the length of one cell then the probability a frame will lose a non-END cell due to a burst error is about $(N-1)P_B = (N-1)9x10^{-8}$.

Random bit errors in the cell header may also cause a cell to be dropped. The probability of this occurring was approximated in equation (4). Thus, a frame loses a non-END cell due to random bit errors with probability $(N-1)\binom{80}{2}P_R^2 = (N-1)\,3x10^{-13}$.

We conclude that the overall probability that a frame at the receiver is missing at least one of its non-END cells is about $(N-1)10^{-6}$, and the dominant cause is congestion. Due to the lost cell, the frame CRC calculation at the destination will essentially produce a random result. Thus, the 30 bit frame CRC will fail to detect the lost cell with probability 2^{-30}. The annual expected frequency of frames with undetected lost non-END cells, per line, is then: $\frac{10^{13}}{N}(N-1)10^{-6}\,2^{-30} \approx 9x10^{-3}$. This does not quite meet our goal of 10^{-3}. Increasing the length of the CRC to 34 bits provides sufficient protection. This is a small increase in the amount of overhead per frame. Thus, we will assume that at least a 34 bit frame CRC should be used.

5.3.2 End Cell Lost

In general, if the END cells of X consecutive frames are lost, then the cells of as many as X+1 frames are merged together. We will consider the simplest case where X equals 1. Using the above assumptions, an END cell is lost with probability: $P_C + P_B + \binom{80}{2} P_R^2 \approx 10^{-6}$. The expected annual frequency of frames per line losing the END cell is then: $\frac{10^{13}}{N}10^{-6}$. The frame CRC of the latter of the two merged frames will be used to check the resulting frame (assuming the CRC is contained in the last cell of a frame). It essentially will be checking random data, so

a 34 bit frame CRC will fail to detect the error with probability 2^{-34}. In the worst case, when N is 2 (N must be at least 2 for a lost cell to result in a merged frame), the expected annual frequency of frames with undetected lost END cells, per line, is: $\frac{10^{13}}{2}\ 10^{-6}\ 2^{-34} = 3\text{x}10^{-4}$.

Instead of relying solely on the CRC to detect lost cells, we could consider adding a frame length field. However, a frame length field does not detect all lost cell scenarios. Assume that the length field is placed in the last cell of the frame. If the beginning of frame A is merged with the end of frame B, and the resulting merged frame contains the same number of cells as frame B originally contained, then the length field will not help detect the error. We can derive an upper bound for the probability of this occurring as follows. Assume frame A originally contains M cells and frame B originally contains N cells. Assume a burst of congestion hits frames A and B such that the last cell of frame A is lost but the last cell of B is not lost. Assume with probability $\frac{1}{\text{M-1}}$ the number of remaining cells in A is *i*, where *i* ranges from 1 to M-1, and with probability $\frac{1}{\text{N}}$ the number of cells remaining in B is *j*, where *j* ranges from 1 to N (if *j* equals N then frame B is unaffected by the congestion). If $\text{M} \le \text{N}$, then the merged frame will contain exactly N cells if there are *i* cells remaining in frame A and N-*i* cells remaining in frame B, for $1 \le i \le \text{M-1}$. If $\text{M} > \text{N}$, then the merged frame will contain N cells if there are *j* cells remaining in frame B and N-*j* cells remaining in frame A, for $1 \le j \le \text{N-1}$. Thus, the probability the merged frame will contain precisely N cells is:

$$\text{if M} \le \text{N: } (\text{M-1})\left(\frac{1}{\text{N}}\ \frac{1}{\text{M-1}}\right) = \frac{1}{\text{N}} \qquad \text{if M} > \text{N: } (\text{N-1})\left(\frac{1}{\text{N}}\ \frac{1}{\text{M-1}}\right) < \frac{1}{\text{N}}$$

Thus, we can upper bound this probability by 1/N. This represents the approximate fraction of merged frames scenarios that can not be detected by a frame length field.

5.4 Detection of Errors in the END Flag

Next, we consider the scenario where a cell arrives at the correct destination but contains an error in its END flag field. The END flag is a one bit flag in the cell header and is thus protected by the cell header CRC. Gaining the protection of the cell header CRC is the major reason we prefer to include the flag as part of the cell header rather than as part of the segment, despite the fact that the flag is not used at the ATM layer.

In order for an error in the END flag to go undetected, the cell header CRC must fail to detect the error. At least three bit errors must occur before the error will go undetected by the CRC, assuming the CRC is in the correction mode. (It is not necessary that one of the three bit errors be in the END flag; the node could make a false 'correction' that results in an errored

END flag.) Thus, the probability the error occurs due to random bit errors and is not caught by the cell header CRC is upper bounded by: $\binom{40}{3} P_R^3 \approx 10^{-20}$. A burst error hitting the cell header could also cause an error in the END flag. We make the worst case assumption that the address field in the header is unaffected by the burst error so that the cell is not misdirected. Assuming the CRC is in the correction mode, the probability the error occurs due to a burst error and is not caught by the cell header CRC is about: $P_B\ 41\ 2^{-8} = 10^{-8}$. Thus, burst errors are the dominant cause.

First consider the scenario where an END flag is changed to a non-END flag. The errored frame will be merged with the following frame, and the CRC of the next frame will be used to check the resulting frame. It will essentially be checking random bits, and, assuming it is 34 bits long, will fail to detect the error with probability 2^{-34}. Thus, the expected annual frequency of frames with undetected END cell to non-END cell transitions, per line, is: $\frac{10^{13}}{N} 10^{-8}\ 2^{-34}$. This equals 6×10^{-6} for the worst case where N equals 1.

If a non-END cell is changed into an END cell, then the frame is essentially split into two frames. Random bits in the 'false' END cell will be interpreted as the frame CRC for the 'first' frame, and thus will appear to be correct with probability 2^{-34}. The frame CRC in the true END cell will only be checking the latter half of the original frame. Thus, this CRC will also fail with probability about 2^{-34}. Thus, overall, the expected annual frequency of frames with undetected non-END cell to END cell transitions per line is: $\frac{10^{13}}{N}(N-1)\ 10^{-8}\ (2)2^{-34} \approx 10^{-5}$.

For either scenario we meet our goal of 10^{-3}.

5.5 Detection of Misdirected Cells

Lastly, we consider the error scenario where a frame contains a stray cell. In section 5.1, we considered preventing misdirected cells; here we consider detecting a misdirected cell. As shown in section 5.1, the most probable cause of a misdirected cell is a burst error hitting the cell header. Due to the presence of the cell header CRC in the four state correction/detection mode, the probability of a cell being misdirected is $P_B\ 2^{-15}$. We make the worst case assumption that every misdirected cell results in one errored frame at the incorrect destination. (Of course, a misdirected cell also results in an errored frame at the correct destination, but we already discussed the lost cell scenario in section 5.3.)

If a frame contains a stray non-END cell, the frame CRC will essentially be checking random bits. A 34 bit CRC will fail to detect the error with probability 2^{-34}. The expected annual frequency of frames containing undetected stray non-END cells, per line, is:

$10^{13}\ P_B\ 2^{-15}\ 2^{-1}\ 2^{-34} = 6x10^{-10}$. The 2^{-1} term is the probability that a stray cell will not have its END flag set after its header is hit by a burst error.

If a frame contains a stray END cell, then the frame is essentially split into two frames. The frame CRC in the stray END cell will be used to check the 'first' frame. The frame CRC in the true END cell will check the 'second' frame. Either CRC will fail with probability about 2^{-34}. Thus, overall, the expected annual frequency of frames containing undetected stray END cells, per line, is: $10^{13}\ P_B\ 2^{-15}\ 2^{-1}(2)\ 2^{-34} = 1x10^{-9}$.

For either scenario we meet our goal of 10^{-3}.

Consider adding a frame length field to the end of a frame to help detect the scenario where a stray cell is accepted as part of a frame at the wrong destination. Assume the stray cell is really an END cell of a frame that has N cells. If it arrives at the wrong destination such that it is accepted as the N^{th} and final cell of a frame, then the length field in the stray cell would fail to detect this scenario (assuming it was not affected by the error that caused the misdirection). We can approximate the likelihood of this event as follows. Assume the misdirected cell belongs to a connection where all frames are comprised of N cells. Thus, with probability 1/N the stray cell is an END cell; assume the burst error does not affect the END flag. Assume it is misdirected to a destination where the frames are comprised of M cells. There is a 1/M chance that the stray cell will arrive before the i^{th} cell of a frame, for $1 \leq i \leq M$. If $N \leq M$, then the stray END cell will be accepted as the N^{th} cell with probability 1/M. If $N > M$, then it can't be accepted as the N^{th} cell. Thus, with these assumptions, the fraction of misdirected cell scenarios that can't be detected by a frame length field can be loosely upper bounded by $\frac{1}{MN}$.

There is another special case to consider. Assume a frame is comprised of just a single cell, and assume this cell is hit by a burst error and misdirected. Assume that only the cell header is affected by the burst error; the remainder of the cell is intact. Assume that the cell arrives at the wrong destination immediately after an END cell, and that its own END flag is still set; thus, it still will appear to be a single cell frame. The frame CRC does not help detect the misdirection since the frame information is intact (the frame CRC doesn't check the cell header). There is no means of detecting the misdirection. In the very worst case where every frame is a single cell frame, the expected annual frequency of this event per line is: $10^{13}P_B\ 2^{-15} \approx 20$.

To add greater protection against this error, we can implicitly include the destination address in the frame when calculating the frame CRC. At the transmitter, the frame CRC is calculated as if the address of the desired destination preceded the frame data. At the receiver, the frame CRC is calculated as if the address of the receiver preceded the frame data. If the destination address is less than or equal to 34 bits, then the CRC will detect the misdirection

with certainty (a CRC of length L can always detect error bursts of less than L bits [5]). If the address is longer, the frame CRC will fail to detect the error with probability about 2^{-34}. (We assume that the incorrect destination address is uncorrelated with the correct destination address; thus, the effect of the misdirection is similar to a burst error hitting the destination field.) Thus, the expected annual frequency of undetected misdirected single cell frames, per line, can be upper bounded by: $10^{13} P_B \, 2^{-15} 2^{-34} = 2 \times 10^{-9}$.

5.6 Summary

From the discussion above, we conclude that the error detection scheme should consist of:

8 bit cell header CRC in four-state correction/detection mode

34 bit frame CRC

Destination address implicitly checked by the frame CRC

The performance of this scheme is summarized in Table 1. The contrast with AAL 5 is discussed in Section 7. We arbitrarily chose N to be 10 for those frequencies that depend on the number of cells per frame. We have not considered out-of-sequence or duplicate cells since these error scenarios are not expected to occur in ATM. However, the frame CRC would provide protection if these errors did occur; the CRC would fail to detect an out-of-sequence or duplicate cell in a frame with probability about 2^{-34}.

TABLE 1

Error Type	Chief Cause	Expected Annual Freq. of Occurrence	Expected Annual Freq. of Undetected Error	
			Our Scheme	AAL 5
Bit Errors in Data	Burst error	9×10^{5}	5×10^{-5}	2×10^{-4}
	Random bit errors	4×10^{7}	2×10^{-11}	2×10^{-11}
Lost non-END Cell	Congestion	10^{7}	6×10^{-4}	2×10^{-11}
Lost END Cell	Congestion			
Length Change		1×10^{6}	6×10^{-5}	2×10^{-12}
No Length Change		1×10^{5}	6×10^{-6}	2×10^{-5}
Error in END Flag	Burst Error	5×10^{5}	10^{-5}	7×10^{-10}
Misdirected Cell	Burst Error			
Length Change		20	2×10^{-9}	4×10^{-14}
No Length Change		.2	2×10^{-11}	5×10^{-11}

6. IMPLEMENTATION

The 34 bit frame CRC should be placed at the very end of the last cell of the frame. The last cell or both the last cell and the second to last cell of the frame may contain less than a complete 48 bytes of information. Thus, there needs to be a pad length field immediately preceding the frame CRC to indicate the number of bytes between the end of the frame data and the beginning of this pad length field. The pad length field should be 6 bits long since the pad length is no longer than the length of one segment (i.e., 48 bytes). The frame CRC should be calculated over the entire frame, including the pad field and the pad length field. This ensures that up to three bit errors in the pad length field are caught with certainty, assuming there are no other errors in the frame. The frame format is shown in Figure 3.

Figure 3 Format of frame with one per-frame CRC.

7. ALTERNATIVE SCHEMES

In the scheme proposed above, we rely on the frame CRC to detect most errors. One drawback to relying solely on a frame CRC is that if the CRC fails to detect a congestion loss, then an undetected error event occurs without there being any type of bit error. A variety of fields could be added to reduce the probability of such an event. However, as we discuss below, even with the addition of these fields, we cannot eliminate the possibility of this occurring.

Also, the analysis presented above largely depends on our estimates of the underlying errors in ATM systems. Our estimates of random bit errors and burst errors are probably conservative. However, we are unsure of whether 10^{-6} is a realistic estimate of the cell loss rate due to congestion; thus, it may be desirable to provide greater protection against cell loss. One option is to increase the length of the frame CRC.

An alternative is to add other fields to detect the scenarios that involve cell loss. For example, we can take advantage of the fact that most scenarios involving cell loss also result in the length of the frame being changed. Thus, we can replace the pad length field by a 16 bit frame length field, and reduce the frame CRC to 32 bits. (The CRC needed to be 34 bits long to protect against lost cells; however, with the addition of the frame length field, the CRC can be reduced to 32 bits.) This increases the amount of overhead by 8 bits per frame. Note that

the AAL 5 error detection scheme consists of a 16 bit frame length field and a 32 bit frame CRC. The performance of this scheme is shown in Table 1.

The overall expected annual frequency of undetected error per line in our proposed scheme and in AAL 5 is approximately the same ($7x10^{-4}$ vs. $2x10^{-4}$). The chief cause of undetected error in our scheme is congestion resulting in non-END cells being dropped. Recall that when calculating the frequency of this error, we assumed cells are lost independently due to congestion, which is an extreme worst case assumption. The chief cause of undetected error in AAL 5 is burst errors resulting in bit errors in the data. In calculating the frequency of this error, we assumed cells are hit independently by short burst errors, which is also an unlikely assumption. Thus, we cannot state definitively which scheme performs better since the performance is tied to the precise nature of the congestion and burst errors.

Note that although AAL 5 provides greater protection against the lost cell scenarios where a length change is involved, it provides less protection against the lost cell scenario where a length change is not involved (i.e., the merged frame scenario discussed in section 5.3.2). Only the frame CRC, which is shorter by 2 bits in AAL 5, provides protection against this latter scenario. Thus, this scheme is really not a safeguard against increases in the congestion rate.

One way to provide greater protection against the merged frame scenario is to include a frame ID field in both the first and last cells of a frame. Assume the length of the frame ID field is F bits. Under most circumstances, there would have to be a bit error in one of the ID fields before a merged frame could go undetected. However, if frames A and B are separated by 2^F frames (i.e., they have the same frame ID), and congestion hits resulting in frames A and B being merged, then the frame ID field does not help detect the error. Also, if the merged frame contains the same number of cells that frame B originally contained, then the frame length field does not help detect the error. Again, we have the situation where we totally rely on the frame CRC to detect the error. Thus, we cannot totally eliminate scenarios where an undetected error event occurs without there being a bit error (unless the frame ID field is large enough that it never wraps around).

Note that the frame length field and frame ID fields do not help detect bit errors in the data. Only the frame CRC helps detect this error. Thus, the extra bits in these alternate schemes could be added to the length of the CRC to decrease the frequency of <u>all</u> undetected errors.

7.1 AAL 3/4 PROPOSAL

Next, we summarize the error protection scheme of AAL 3/4. Figures 4 and 5 show the format of the frame and segment in this proposal.

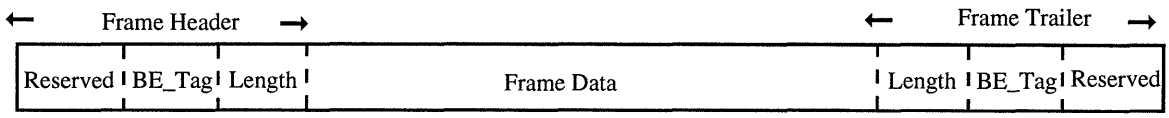

Figure 4 Format of frame in AAL 3/4.

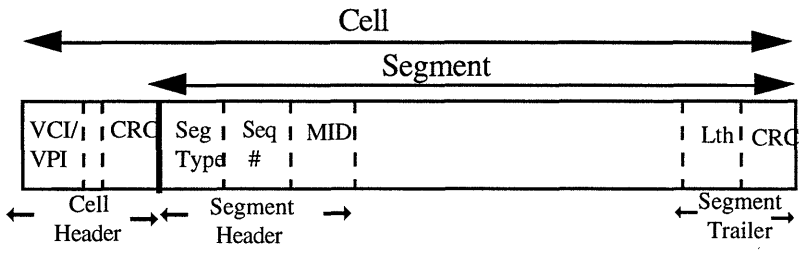

Figure 5 Format of segment in AAL 3/4. The ATM layer adds a 5 byte header to the segment to form a cell.

There is a 16 bit frame length field in both the frame header and trailer to protect against lost cells. There is an 8 bit frame sequence number, referred to as the Begin/End Tag (BE_Tag), in both the frame header and trailer to protect against merged frames. The segment header contains a two bit segment type field that indicates whether the cell is the first, a middle, or last cell of the frame. A four bit sequence number in the segment header helps protect against lost cells. There is also a ten bit message ID field in the segment header that is used for connectionless traffic. The segment trailer includes a 6 bit length field to indicate the number of bytes contained in the segment. Each segment, except for the last segment in the frame, is expected to contain 48 bytes of information (including header and trailer); thus, the segment length field is unnecessary. Finally, there is a 10 bit segment CRC.

The major problem with the AAL 3/4 scheme is that a per-segment CRC is used rather than a per-frame CRC. A per-segment CRC does not help detect lost cells, misdirected cells, or merged frames. This necessitated the addition of fields such as the per-segment sequence number and the Begin/End Tag. The performance of this scheme very much depends on the characteristics of burst errors in the system. If we assume that burst errors always affect a large number of cells, and that all cells that are hit by the burst will contain completely random bits, then the AAL 3/4 scheme provides sufficient protection. If we make the same worst case assumptions that we did in section 5.2, that bursts are very short, then we find that we have a probability of 2^{-10} of not detecting burst errors that affect only the frame data. As shown in section 5.2, this does not provide sufficient protection. Since we can't be sure exactly what type of burst errors to expect, it makes more sense to use an error detection scheme that is powerful over a wider range of errors. Thus, a per-frame CRC is preferred over a per-segment CRC.

There are other scenarios that point out the weakness of not having a frame CRC. For example, consider the scenario where a cell is misdirected and is accepted as the last cell of a frame in place of the correct end cell. Assume the misdirection is caused by a burst error and the burst is short enough so that only the cell header is affected. Since there is no frame CRC, the AAL 3/4 scheme relies on the segment sequence number and the Begin/End Tag to catch the error. Although this error event does not pose major problems, it points out how little protection there is against some of the error scenarios when a per-frame CRC is not present. We did not include the performance of the AAL 3/4 scheme in Table 1 since there are many additional error scenarios that arise due to errors in the control fields (e.g., errors in the segment type).

The AAL 3/4 scheme is obviously less efficient than our proposed scheme and the scheme proposed in AAL 5. Assume a frame is comprised of 10 cells. The amount of overhead in the various schemes is:

Our Proposed Scheme: 120 bits AAL 5 Proposal: 128 bits AAL 3/4 Proposal: 464 bits

If the number of cells per frame is large, then the overhead per cell is roughly 8 bits in our proposed scheme and in AAL 5, and 40 bits per cell in AAL 3/4. Note that a large portion of ATM traffic is expected to consist of video images, which typically involve large frames.

We conclude that the AAL 3/4 scheme requires more overhead and provides less protection than our proposed scheme and AAL 5.

8. CONCLUSIONS

For the connection oriented service class of ATM, a 34 bit frame CRC should provide sufficient protection in attaining our goal of no more than one undetected errored frame per receiver per year. This solution should be very robust. We also showed that a four-state correction/detection option for the cell header CRC is a simple method of reducing the probability of misdirected data. Our proposed scheme is more effective and efficient than the CCITT AAL 3/4 proposal, but is similar to the CCITT AAL 5 proposal.

REFERENCES

[1] CCITT Recommendation I.363, "B-ISDN ATM Adaptation Layer (AAL) Specification, Geneva, 1992.

[2] CCITT Study Group XVIII/8-5, "AAL Type 5, Draft Recommendation Text For Section 6 of I.363," Copenhagen, October, 1992.

[3] Bellcore, "Preliminary Report on Broadband ISDN Transfer Protocols," SR-NWT-001763, Issue 1, December 1990.

[4] Vanstone, S., and van Oorschoot, P., *An Introduction to Error Correcting Codes with Applications*, Kluwer Academic Publishers, Boston, 1989.

[5] Bertsekas, D. and Gallager, R., *Data Networks*, Second Edition, Englewood Cliffs, NJ: Prentice-Hall, Inc., 1992.

[6] Dravida, S., Cheng, Y., and Saksena, V.R., "Error performance of broadband ISDN networks," submitted to *IEEE Transactions on Communications*, 1991.

[7] CCITT Recommendation I.432, "B-ISDN User-Network Interface - Physical Layer Specification", Matsuyama, December, 1990.