Optical Network Packet Error-Rate due to Physical Layer Coding

Andrew W. Moore, *Member, IEEE,* Laura B. James, *Member, IEEE,* Madeleine Glick, *Member, IEEE,* Adrian Wonfor, *Member, IEEE,* Richard Plumb, *Member, IEEE,* Ian H. White, *Fellow, IEEE,* Derek McAuley, *Member, IEEE* and Richard V. Penty, *Member, IEEE*

***Abstract*— A physical layer coding scheme is designed to make optimal use of the available physical link, providing functionality to higher components in the network stack. This paper presents results of an exploration of the errors observed when an optical Gigabit Ethernet link is subject to attenuation. The results show that some data symbols suffer from a far higher probability of error than others. This effect is caused by an interaction between the physical layer and the 8B/10B block coding scheme. We illustrate how the application of a scrambler, performing *data-whitening*, restores content-independent uniformity of packet-loss. We also note the implications of our work for other (N,K) block-coded systems and discuss how this effect will manifest itself in a scrambler-based system. A conjecture is made that there is a need to build converged systems, with the combinations of physical, data-link, and network layers optimised to interact correctly. In the mean time, what will become increasingly necessary is both an identification of the potential for failure and the need to plan around it.**

Topic Keywords: Optical Communications, Networks, Codecs, Systems engineering, Data Communications.

I. Introduction

MANY modern networks are constructed as a series of layers. The use of layered design allows for the modular construction of protocols, each providing a different service, with all the inherent advantages of a module-based design. Network design decisions are often based on assumptions about the nature of the underlying layers. For example, design of an error-detecting algorithm, such as a packet checksum, will be based upon premises about the nature of the data over which it is to work and assumptions about the fundamental properties of the underlying communications channel over which it is to provide protection.

Yet the nature of the modular, layered design of network stacks has caused this approach to work against the architects, implementers and users. There exists a tension between the desire to place functionality in the most appropriate sub-system, ideally optimised for each incarnation of the system, and the practicalities of modular design intended to allow independent developers to construct components that will inter-operate with each other through well-defined interfaces. However, past experience has lead to assumptions being made in the construction or operation of one layer's design that can lead to incorrect behaviour when combined with another layer. There are numerous examples describing the problems caused when layers do not behave as the architects of certain system parts expected. An example is the re-use of the 7-bit digitally-encoded voice scrambler for data payloads [1], [2]. The 7-bit scrambling of certain data payloads (inputs) results in data that is (mis)identified by the underlying SONET [3] layer as codes belonging to the control channel rather than the information channel.

It is our conjecture that such layering, while often considered a laudable property in computer communications networks, can lead to irreconcilable faults due to differences in such fundamental measures as the number and nature of errors in a channel, and a misunderstanding of the underlying properties or needs of an overlaid layer.

While the use of layering leading to undesirable side-effects has been observed in the past [4], this paper focuses upon data-integrity issues that arise from the specific interactions between the physical, data-link, and network layers. We also note how the evolution of new technologies driving speed, availability, etc., contribute to the problem of incompatible layering.

Outline

Section II describes our motivations for this work including a summary of research directions for optical packet systems and the implications of the limits on the quantity of power useable in optical networks.

We present a study of the 8B/10B block-coding system, as used in Gigabit Ethernet [5], the interaction between an (N,K) block code, an optical physical layer, and data transported using that block code in Section III. In Section IV we document our findings on the reasons behind the interactions observed.

As an illustration of how these effects may be overcome, Section V presents results for a scrambler used in combination with the 8B/10B codec.

Further to our experiments with Gigabit Ethernet, in Section VI we illustrate how the issues we identify have ramifications for systems with increasing physical complexity and

A. W. Moore is with the Computer Laboratory, University of Cambridge.

L. B. James, R. G. Plumb, A. Wonfor, I. H. White and R. V. Penty are with the Center for Photonic Systems, Department of Engineering, University of Cambridge.

M. Glick and D. McAuley are with Intel Research, Cambridge.

Andrew Moore acknowledges the Intel Corporation's generous support of his research fellowship and Laura James thanks EPSRC & Marconi for their support of her PhD research. Contact author: andrew.moore@cl.cam.ac.uk

also note these issues as they relate to the coding schemes employed in SONET. Section VII details our conclusions from this work.

II. Motivations

A. Optical Networks

Current work in all areas of networking has led to increasingly complex architectures: our interest is focused upon the field of optical networking, but this is also true in the wireless domain. Our exploration of the robustness of network systems is motivated by the increased demands of these new optical systems.

Wavelength Division Multiplexing (WDM) is a core technology in the current communications network. To take advantage of higher capacity developments at the shorter timescales relevant to the local area network, as well as system and storage area networks, packet switching and burst switching techniques have seen significant investigation [6], [7].

Examples of new, complex, optical architectures that incorporate a large number of both active and passive optical components include those based upon Optical Packet Switching (OPS) for high speed, low latency computer networking [8]. One example system is the *Data Vortex* prototype, designed as a specialist interconnect for future super-computers [9].

Our own prototype OPS for the local-area network uses a multi-wavelength optical data path end to end, with a switching system based upon semiconductor optical amplifiers (SOAs) [10], [11]. In the current version of the this system, each wavelength carries data at 1.25Gbps, using 8B/10B coding. As part of this work we recognise that the need for higher data-rates and designs with larger numbers of optical components are forcing us toward what traditionally have been technical limits.

Further to the optical-switching domain, there have been changes in the construction and needs of fibre-based computer networks. In deployments containing longer runs of fibre using large numbers of splitters for measurement and monitoring as well as active optical devices, the overall system loss may be greater than in today's point-to-point links and the receivers may have to cope with much-lower optical powers. Increased fibre lengths used to deliver Ethernet services, e.g., Ethernet in the first mile [12], along with a new generation of switched optical networks, are examples of this trend.

Additionally, we are increasingly impacted by operator practise. For example, researchers have observed that up to 60% of faults in an ISP-grade network are due to optical events [13]: defined as ones where it was assumed errors results directly from operational faults of in-service equipment. While the majority of these will be catastrophic events (e.g., cable breaks), a discussion with the authors of [13] allow us to speculate that a non-trivial percentage of these events may be due to the issues of layer-interaction discussed in this paper.

B. The Power Problem

If all other variables are held constant an increase in bandwidth will require a proportional increase in transmitter power. However, fibre nonlinearities impose limitations on the maximum optical power able to be used in an optical network. Subsequently, we maintain that a greater understanding of the low-power behaviour of coding schemes will provide invaluable insight for future systems.

For practical reasons including availability of equipment, its wide deployment, tractability of the problem-space and and well documented behaviour, as well as its relevance to our own optical networking project [11], we concentrate upon the 8B/10B codec.

C. 8B/10B Block Coding

The 8B/10B codec, originally described by Widmer & Franaszek [14], is widely used. This scheme converts 8 bits of data for transmission (ideal for any octet-orientated system) into a 10 bit line code. Although this adds a 25% overhead, 8B/10B has many valuable properties; a transition density of at least 3 per 10 bit code group and a maximum run length of 5 bits for clock recovery, along with virtually no DC spectral component. These characteristics also reduce the possible signal damage due to jitter, which is particularly critical in optical systems, and can also reduce multimodal noise in multimode fibre connections.

This coding scheme is royalty-free, well understood, and sees current use in a wide range of applications. In addition to being the standard Physical Coding Sublayer (PCS) for Gigabit Ethernet [5], it is used in the Fibre Channel system [15]. This codec is also used for the 800Mbps extensions to the IEEE 1394 / Firewire standard [16], and 8B/10B is the basis of coding for the electrical signals of the PCI Express standard [17].

The 8B/10B codec defines encodings for data octets and control codes which are used to delimit the data sections and maintain the link. Individual codes or combinations of codes are defined for Start of Packet, End of Packet, line Configuration, and so on. Also, Idle codes are transmitted when there is no data to be sent to keep the transceiver optics and electronics active. The Physical Coding Sublayer (PCS) of the Gigabit Ethernet specification [5] defines how these various codes are used.

Individual ten-bit code-groups are constructed from the groups generated by 5B/6B and 3B/4B coding on the first five and last three bits of a data octet respectively. During this process the bits are re-ordered, such that the last bits of the octet for transmission are encoded at the start of the 10-bit group. This is because the last 5 bits of the octet are encoded first, into the first 6 bits of code, and then the first 3 bits of the octet are encoded to the final 4 transmitted bits. Some examples are given in Table I; the running disparity is the sign of the running sum of the code bits, where a one is counted as 1 and a zero as -1. During an Idle sequence between packet transmissions, the running disparity is changed (if necessary) to -1 and then maintained at that value. Both control and data codes may change the running disparity or may preserve its existing value; examples of both types are shown in Table I. The code-group used for the transmission of an octet depends upon the existing running disparity – hence the two alternative codes given in the table. A received code-group is compared

TABLE I
EXAMPLES OF 8B/10B CONTROL AND DATA CODES

Type	Octet	Octet bits	Current RD -	Current RD +	Note
data	0x00	000 00000	100111 0100	011000 1011	preserves RD value
data	0xf2	111 10010	010011 0111	010011 0001	swaps RD value
control	K27.7	111 11011	110110 1000	001001 0111	preserves RD value
control	K28.5	101 11100	001111 1010	110000 0101	swaps RD value

against the set of valid code-groups for the current-receiver running disparity, and decoded to the corresponding octet if it is found. If the received code is not found in that set, the specification states that the group is deemed invalid. In either case, the received code-group is used to calculate a new value for the running disparity. A code-group received containing errors may thus be decoded and considered valid. It is also possible for an earlier error to throw off the running disparity calculation causing a later code-group may be deemed invalid because the running disparity at the receiver is no longer correct. This can propagate the effect of a single bit error at the physical layer. Line coding schemes, although they handle many of the physical layer constraints, can introduce problems. In the case of 8B/10B coding, a single bit error on the line can lead to multiple bit errors in the received data byte. For example, with one bit error the code-group D0.1 (current running disparity negative) becomes the code-group D9.1 (also negative disparity); these decode to give bytes with 4 bits of difference. In addition, the running disparity after the code-group may be miscalculated, potentially leading to future errors. There are other similar examples in [5].

III. EXPERIMENTAL METHOD

We contrast two commonly used metrics: bit-error-rate, as used to describe the physical layer performance, and packet-error-rate: a measurement of network-application performance.

A. Test Environment

We investigate these effects using Gigabit Ethernet equipment on optical fibre, (1000BASE-X [5]) under conditions where the received power is sufficiently low as to induce errors in the Ethernet frames. We assume that while the Functional Redundancy Check (FRC) mechanism within Ethernet is sufficiently strong to catch the errors, the dropped frames and resulting packet loss will result in a significantly higher probability of packet errors than the norm for certain hosts, applications and perhaps users.

We used 1000BASE-ZX Gigabit Ethernet transceivers. The ZX, a Cisco proprietary extension to the official IEEE standard, operates at 1550nm.

In our main test environment an optical attenuator is placed in one direction of a Gigabit Ethernet link. A traffic generator feeds a Fast Ethernet link to an Ethernet switch, and a Gigabit Ethernet link is connected between this switch and a traffic sink and tester (Figure 1). An optical isolator and the variable optical attenuator are placed in the fibre in the direction from the switch to the sink. We had previously noted interference due to reflection and the isolator allows us to remove this aspect from the results.

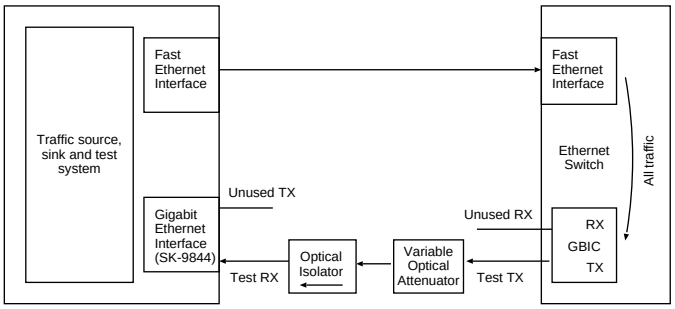


Fig. 1. Main test environment

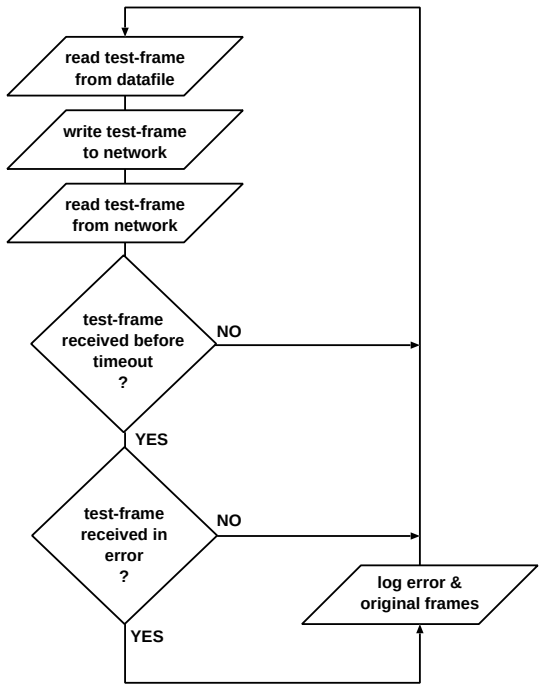


Fig. 2. Flowchart of real-time software

A packet capture and measurement system is implemented within the traffic sink using an enhanced driver for the SysKonnect SK-9844 network interface card (NIC). Among a number of additional features, the modified driver allows application processes to receive error-containing frames that would normally be discarded. As well as purpose-built code for the receiving system, we use a special-purpose traffic generator and comparator which are combined into one real-time software module (Figure 2). This system, based upon *tcpfire* [18], transmits pre-constructed test data in tcpdump/pcap-format. Transmitted frames are compared to their received versions and if they differ, both original and error frames are

stored for later analysis.

A range of receiver optical powers (equivalent to varied bit error rates) were used for testing. Even at powers slightly below the receiver sensitivity, the equipment used at no point ceased to send packets of data to the host computer, and did not indicate that the optical power was too low or that the receiver was suffering errors.

1) Octet Analysis: Each octet for transmission has been encoded by the Physical Coding Sublayer of Gigabit Ethernet using 8B/10B into a 10 bit code-group or *symbol*, and we analyse these for frames which are received in error at the octet level. By comparing the two possible transmitted symbols for each octet in the original frame to the two possible symbols corresponding to the received octet we can deduce the bit errors which occurred in the symbol at the physical layer (Figure 3). In order to infer which symbol was sent and which received, we assume that the combination giving the minimum number of bit errors on the line is most likely to have occurred. This allows us to determine the line errors which most probably occurred.

Various types of symbol damage may be observed. One of these is the single-bit error caused by the low signal to noise ratio at the receiver. A second form of error results from a loss of bit clock causing *smeared* bits: where a subsequent bit is read as having the value of the previous bit. A final example results from the loss of symbol clock synchronisation. This can lead to the symbol boundaries being misplaced, so that a sequence of several symbols, and thus several octets, will be incorrectly recovered. Some of these error types should have been detected by the Physical Coding Sublayer of Gigabit Ethernet; we postulate that the hardware implementations we have observed do not fully comply with the specification in terms of their decoding algorithms, and/or their handling of error signals.

2) Real Traffic: Results presented here are conducted either with the test-frames indicated or with real network traffic referred to as the *day-trace*. This network traffic was captured from the interconnect between a large research institution and the Internet over the course of two working days [19]. We consider it to contain a representative sample of network traffic for an academic/research organisation of approximately 150 users.

Other traffic tested included *pseudo-random data*, consisting of a sequence of frames of the same number and size as the *day-trace* data — preserving packet-size characteristics — although each is filled with a stream of octets whose values were drawn from a pseudo-random number generator.

3) Bit Error Rate Measurements: For our BER measurements, a directly modulated 1548nm laser was used. The optical signal was then subjected to variable attenuation before returning via an Agilent Lightwave (11982A) receiver unit into the BERT (Agilent parts 70841B and 70842B). The BERT (Bit Error Rate Test-kit) was programmed with a series of bit sequences, each corresponding to a frame of Gigabit Ethernet data encoded as it would be for the line in 8B/10B. Purpose-built code is used to convert a frame of known data into the bit-sequence suitable for the BERT. The bit error rates for these packet bit sequences were measured at a range of attenuation values, using identical BERT settings for all frames (e.g., 0/1 thresholding value).

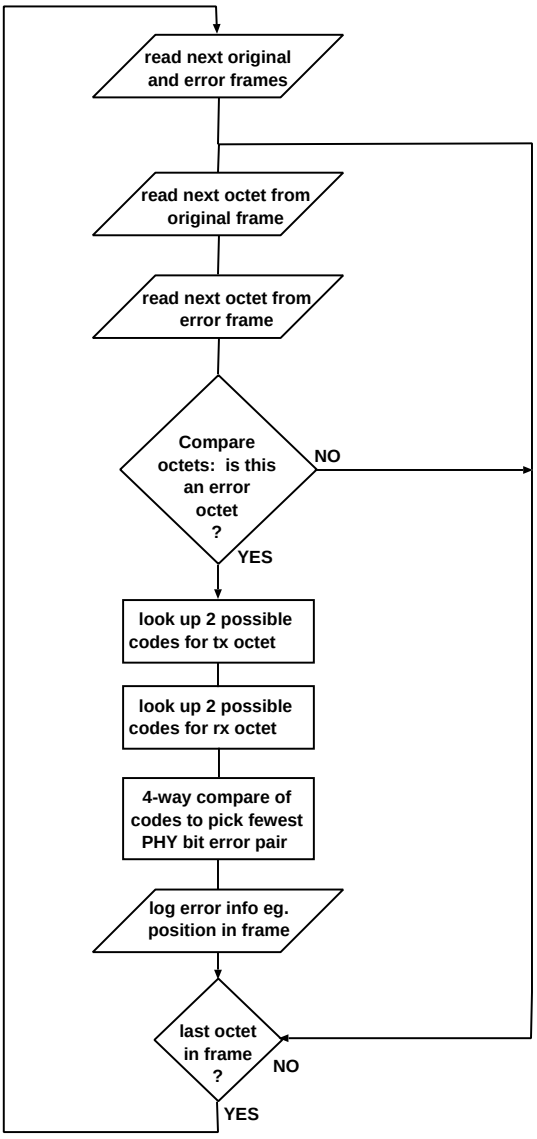


Fig. 3. Flowchart of octet analysis software

Our experiences using this test environment identified that a uniformly-distributed set of random data, after encoding with 8B/10B will not suffer code-errors with the same uniformity. Some octets are much more subject to error than others: error *hot-spotting*. We considered that the 8B/10B coding was actually the cause of this non-uniformity. Our results, [20], clearly showed that the relationship between bit-error rate versus attenuation could not offer a prediction of the outcome for packet-error rate versus attenuation. This specific result allowed us to conclude the relationship was non-deterministic and led to our investigation of the impact upon physical-layer errors the coding scheme had when those errors would be represented in the data-link layer.

Further sets of wide-ranging experiments allowed us to conclude that Ethernet frames containing a given octet of certain value were up to 100 times more likely to be received in error (and thus dropped), when compared with a similar-

sized packet that did not contain such octets [21].

IV. Results and Discussion

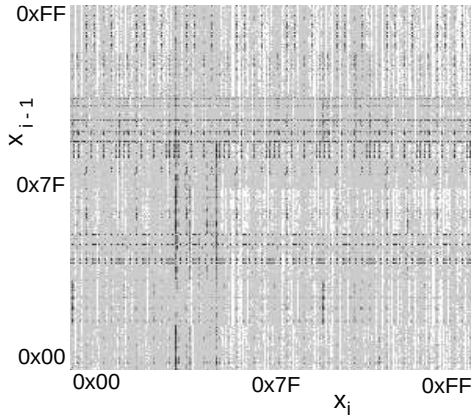
A. Effects on data sequences

We have found that individual errored octets do not appear to be clustered within frames but are independent of each other. However, we are interested in whether earlier transmitted octets have an effect on the likelihood of a subsequent octet being received in error. We had anticipated that the use of running-disparity in 8B/10B would present itself as correlation between errors in current codes and the value of previous codes.

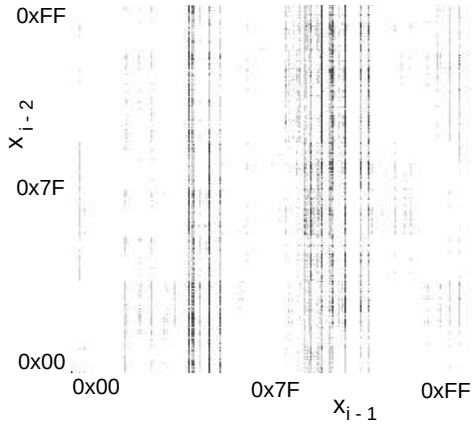
We collect statistics on how many times each transmitted octet value is received in error, and also store the sequence of octets transmitted preceding this. The error counts are stored in 2D matrices (or histograms) of size 256×256, representing each pair of octets in the sequence leading up to the errored octet: one for the errored octet and its immediate predecessor, one for the predecessor and the octet before that, and so on. We normalise the error counts for each of these histograms by dividing by the matrix representing the frequency of occurrence of this octet sequence in the original transmitted data. We then scale each histogram matrix so that the sum of all entries in each matrix is 1.

Figure 4(a) shows the error frequencies (darker values represent more errors) for the "current octet" X_i (the correct transmitted value of octets received in error), on the x-axis, versus the octet which was transmitted before each specific errored octet, X_{i-1}, on the y-axis. Figure 4(b) shows the preceding octet and the octet before that: X_{i-1} vs X_{i-2}. Vertical lines in Figure 4(a) are indicative of an octet that is error-prone independently of the value of the previous octet. In contrast, horizontal bands indicate a correlation of errors with the value of the previous octet.

It can be seen from Figure 4 that while correlation between errors and the value in error, or the immediately previous value, are significant, beyond this there is no significant correlation. The equivalent plot for X_{i-2} vs. X_{i-3} produces a featureless white square.



(a) Error counts for X_i vs. X_{i-1}



(b) Error counts for X_{i-1} vs. X_{i-2}

Fig. 4. Error counts for pseudo-random data octets, darker values represent more errors

B. 8B/10B code-group frequency components and their effects

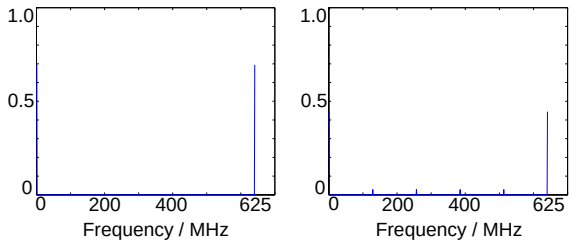
It is illustrative to consider the octets which are most subject to error, and the 8B/10B codes used to represent them. In the pseudo-random data, the following ten octets give the highest error probabilities (independent of the preceding octet value): 0x43, 0x8A, 0x4A, 0xCA, 0x6A, 0x0A, 0x6F, 0xEA, 0x59, 0x2A. It can be seen that these commonly end in A, and this causes the first 5 bits of the code-group to be `01010`. The octets not beginning with this sequence in general contain at least 4 alternating bits. Of the ten octets giving the lowest error probabilities (independent of previous octet), which are 0xAD, 0xED, 0x9D, 0xDD, 0x7D, 0x6D, 0xFD, 0x2D, 0x3D and 0x8D, the concluding D causes the code-groups to start with `0011`.

Fast Fourier Transforms (FFTs) were generated for data sequences consisting of repeated instances of the code-groups of 8B/10B. Examining the FFTs of the code-groups for the high error octets, Figures 5(a) and 5(b), for example, the peak corresponding to the base frequency (625MHz, half the baud rate) is pronounced in most cases, although there is no such feature in the FFTs of the code-groups of the low error octets (Figures 5(c) and 5(d)).

The pairs of preceding and current octets leading to the greatest error (which are most easily observed in Figure 4) give much higher error probabilities than the individual octets. The noted high error octets (e.g. 0x8A) do occur in the top ten high error octet pairs and normally follow an octet giving a code-group ending in `10101` or `0101`, such as 0x58, which serves to further emphasise that frequency component.

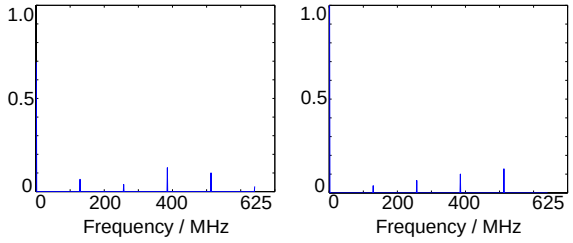
The 8B/10B codec defines both data and control encodings, and these are represented on a 1024x1024 space in Figure 6(a), which shows valid combinations of the current code-group (C_i) and the preceding one (C_{i-1}). The regions of valid and invalid code-groups are defined by the codec's use of 3B/4B and 5B/6B blocks (Section II-C).

In Figure 6(a) the octet errors found in the *day-trace* have



(a) FFT of code-group for high error octet 0x4A

(b) FFT of code-group for high error octet 0x0A



(c) FFT of code-group for low error octet 0xAD

(d) FFT of code-group for low error octet 0x9D

Fig. 5. Contrasting FFTs for a selection of code-groups

been displayed on this codespace, showing the regions of high error concentration for real Internet data. It can be seen that these tend to be clustered and that the clusters correspond to certain features of the code-groups. Two groups of clusters of equal area have been ringed, those that are indicated as $C_i = 0011\ldots$ represent those codes with a low-error suffix. In contrast the ringed values indicated as $C_i = 010101\ldots$ indicates the error-prone symbols with a suffix of 0xA.

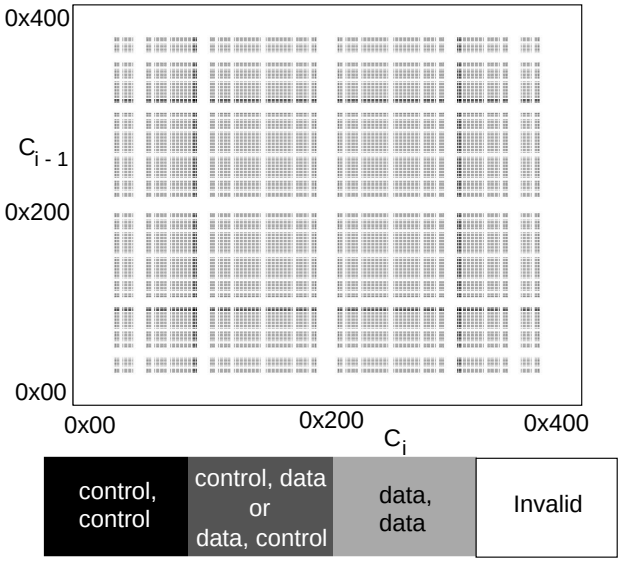
C. Transceiver Effects

It is well known that in a directly modulated optical source it is possible that bandwidth limitations can cause *single ones* to achieve slightly less amplitude than a run of multiple ones. In normal operation, this resultant slight eye closure has no effect on the error rate of the received signal. Figure 7 illustrates this effect of slight eye-closure due to the data-pattern in an operating Gigabit Ethernet link.

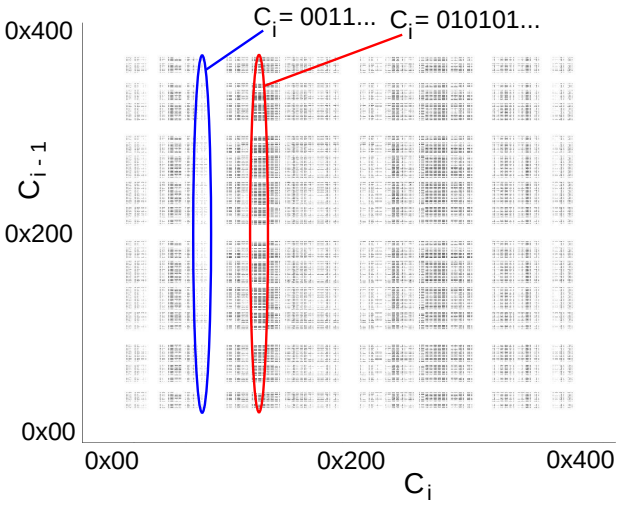
Despite this eye-closure, error-free operation is achieved at a received power significantly above the receiver sensitivity. However, as the received power is reduced toward the sensitivity of the optical receiver it is the *single ones*, e.g. `010101` which produce errors first, as these are of lower amplitude than the *multiple ones*, e.g., `110011`. In addition to optical issues of data-pattern, the packaging requirements imposed in the electrical domain can exacerbate this effect. These broadband limitation effects will be much more significant at the increased modulation rates required for 10 Gbps Ethernet.

V. Whitening 8B/10B

An alternative to (N,K) block codes such as 8B/10B, scrambling also provides a process of encoding digital “1”s and “0”s



(a) Valid $C_{(i-1)}, C_i$ pairs



(b) Errors using *day-trace* as a function of code-groups

Fig. 6. The codebook for 8B/10B represented on a 1024x1024 space

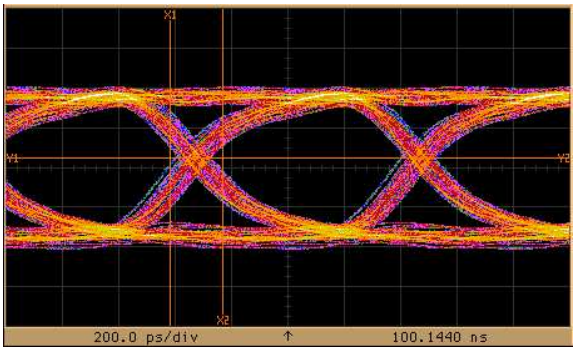


Fig. 7. Eye diagram for an 8B/10B-based Gigabit Ethernet link.

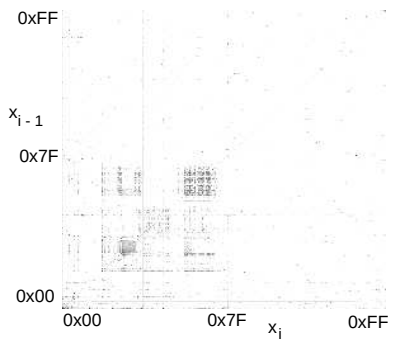


Fig. 8. Frequency of occurrence of previous and current octets in the *day-trace*

onto a line in such a way that provides an adequate number of transitions, and a given "1"s density requirement. A number of communications standards use scramblers; one example is SONET, which uses a 7-bit scrambler by default or a, higher-grade, 44-bit scrambler for data payloads. Another example is the 10 Gbps Ethernet standard 10GBASE-LR which uses a 64B/66B encoding system [22].

Additionally, the use of scramblers to pre-process data prior to coding, referred to as *data-whiteners*, is common. The IEEE 802.15.4 spread-spectrum wireless personal area network (WPAN) [23] specifies a whitener to suppress the power spectral density. A further example is the 800 Mbps Firewire/IEEE 1394b specification which uses a data-whitener to normalise data and improve the performance of the 8B/10B codec used in that system.

We used an implementation of the 64B/66B scrambler from the 10 Gbps Ethernet standard to *whiten* the *day-trace* frames. From Figure 8 we know that this real internet data is non-uniform, concentrated on certain octet values. Clearly this will exacerbate the non-uniform error patterns noted in Section III, as some of the octet sequences most subject to error also occur in the most frequently transmitted *day-trace* regions. By whitening the data before transmission, we expect to spread the octets transmitted over the entire available octet space, such that the 8B/10B codebook is fully utilised, and high error code-groups are sent no more often than low error ones. This also means that when a high error code-group or code-group sequence is received in error, it is not always the same transmitted data pattern which is received in error, restoring uniformity assumptions required for the FRC in use by the data-link layer.

The scrambler is run continuously, rather than restarting frame-by-frame. As a *shim*-layer implemented between data-link layer and network-layer, our implementation whitens only the data of the Ethernet payloads, not the packet headers or the FRC.

We find that our whitened *day-trace* contains all possible octet pairs at frequencies similar to the pseudo-random frames, so the varied characteristics of the *day-trace* have been successfully whitened by the scrambler.

When we compare the octet errors in our attenuated, 8B/10B-encoded system for these new, whitened frames, we see that it follows a similar pattern to that for the pseudo-random frames. Notably our results display patterned errors (*hot-spotting*) in the scrambled data, but following descrambling no measurable correlation is present between payload contents and data in error. We have therefore successfully improved the uniformity of the data errors with respect to the actual transmitted data.

The whitening scheme used removes the non-uniformity of the data errors due to concentrations of transmitted data at certain octet values, but the overall loss level is unchanged as this is due to the coding scheme and physical devices used. While not specifically useful at reducing the level of loss, the use of a scrambler has removed the occurrence of *hot-spotting* within the payload data. While the error-prone octets still exist, by encoding with the scrambler and biases in input data are removed. By removing the *hot-spotting*, the data-dependent errors, we have also restored the underlying uniformity of error assumed by the FRC algorithm and thus improved the data-integrity but removing bias in the face of error.

The use of a stream scrambler has lead to some improvement, but it should be noted that scramblers can react poorly to bad inputs; this issue is discussed in Section VI.

We have demonstrated that the addition of a payload whitening scheme has restored the underlying assumption of uniform errors at the physical layer, and therefore it is anticipated that higher-layer functionality will not suffer. Since networks must often continue to work with legacy layers which cannot be changed or redesigned, the ability to work around their characteristics through the use of *shim* layers, such as the scrambler we illustrate here, becomes increasingly necessary.

VI. Implications

Gigabit Ethernet, when operated according to the specification, is a robust and effective standard. Our results illustrate that if degradation of a Gigabit Ethernet link occurs, then errors can be expected to not be uniform at the higher layers.

In future networks (Section II-A) the low power levels at the receiver might not be well-suited to bit-by-bit detection and decoding, as used by standard 8B/10B systems. The issues described here apply equally to other (N,K) block coded systems, where similar interactions between coding and physical layer pattern-dependent error probabilities occur.

In Section III we documented the occurrence of error *hot-spots*: data and data-sequences with a higher probability of error. In addition to increasing the chances of frame-discard due to data-contents, the occurrence of such *hot-spots* also has implications for higher level network protocols. Frame-integrity checks, such as a cyclic redundancy check, assume that there will be a uniformity of errors within the frame, justifying detection of single-bit errors with a given precision. While Jain [24] demonstrates that the FRC as used in Ethernet is sufficiently strong as to detect all 1, 2 and 3 bit errors for frames up to 8 KBytes in length, problems may be encountered for certain combinations of errors above this. Recall that in Section II-C we noted that many single-bit errors on the physical layer will translate into multi-bit errors following decoding by the PCS.

A. Scrambler Issues

As stated earlier, a primary reason for enforcing a given density of "1"s — in common with all coding schemes — is a requirement for timing recovery or network synchronisation. However, other factors such as automatic-line-build-out (ALBO), equalisation, and power usage are affected by "1"s density. Early packet-over-SONET specifications [1] inadvertently permitted malicious users to generate packets with bit patterns that could create SONET density synchronisation problems by replicating the sequences of bits identified as frame alignment. The solution to this was to provide a more secure mechanism for payload scrambling. As noted in Malis & Simpson [2], this was the addition of payload scrambling using an $x^{43} + 1$ self-synchronous scrambler, as is also used when transmitting ATM over SONET. This scrambler reduces the chance of malicious (or accidental) emulation of control sequences to less than 1 in 9^{16}.

However, because all SONET headers must have interoperablity, the scrambler used for ATM over SONET, and described in Malis & Simpson [2], only applies to the payload of the SONET frame and not the header. The SONET headers are restricted to using the 7-bit scrambler: $1 + x^6 + x^7$. This scrambler, limited to 7-bits in length, has a repeat-rate of $2^n - 1 = 127$ cycles. Such a 7-bit scrambler was considered sufficient for voice data, but we note a number of unanticipated long-term implications of a scrambler of this length.

While such a short scrambler has not shown problems that immediately identify it as the cause, the 7-bit coding of headers has become a necessary constant for SONET regardless of the data-rate. Hence this built-in limitation may be expected to cause similar, unpredictable interactions as those described in Section IV-C. We anticipate this may lead to data input-specific errors similar to those we identify using the 8B/10B codec and encourage the research community to investigate this space further.

The 8B/10B scheme has an elegant balance between clock and data recovery ability and the cost and efficiency of its implementation. Whether or not a scrambler should be added to a system is a tradeoff between implementation complexity and functionality, and depends on the network and application in question.

B. Network/Transport Layer Issues

Up until now we have concentrated upon the interaction between the physical layer and the data-link layer, such as that embodied in 1000BASE-X. We briefly note the interaction that data-link layer effects may have with the network and transport layer.

In James *et al.* [25] we highlighted the non-uniform distribution of packet errors that result from an interaction between the physical coding conditions, the 8B/10B coding scheme, and particular data to be transported through the network. That work identified that certain data-values had a substantially higher probability of being received in error, which resulted in packets with those payloads being discarded with a higher-than-normal probability. This non-uniformity becomes an issue when the designers of higher level network protocols expect otherwise, regardless of the actual error rate [26].

An analysis of the contents of *day-trace* data along with other data derived as part of our network-monitoring work allows us to conclude that in addition to (user) data-payloads the error-concentrating effects will cause a significant level of loss due to the network and transport-layer header contents. In one hypothetical case, if a user on a machine with an IP address that consisted of several high-error-rate octets their data will be at a proportionally higher risk of being corrupted and discarded.

Further, the occurrence of error *hot-spots* has other ramifications. Stone *et al.* [27] discuss the impact this has for the checksum of TCP; they found that error-conditions exist that could cause data to be consider valid after examination of the TCP checksum despite errors being present in the data itself. These results may call into question our assumption that only increased packet-loss will be the result of the error *hot-spots*. Instead of just lost packets, Stone *et al.* noted certain "unlucky" data would rarely have errors detected.

Various techniques could be employed to enhance the ability of a system operating in a low-power state to recover error-free data; forward error correction (FEC) would be one of these (and indeed is incorporated into the specification for Ethernet in the First Mile [12].

VII. Conclusions

Examining the 8B/10B code, used in Gigabit Ethernet and elsewhere, we have documented the form and cause of failures that occur in the low-power regime, inducing, at best, poor performance and, at worst, undetected errors that may focus upon specific networks, applications and users. The errors observed in 8B/10B encoded data in a low-power regime are not uniform. Section VI-B and the references therein indicate that uniformity has been assumed in the past. Some packets will suffer greater loss rates than the norm. This content-specific effect difficult to diagnose because it occurs without a total failure of the network, and will distort the frame error rate relative to frame content.

We note the reasons for the pattern-related failure modes are a combination of layer-related effects. Alongside the documented *hot-spotting* of errors due to the 8B/10B block-code, we also note the well-known fact that physical layer errors are pattern dependent. This is due to bandwidth limitations in the physical transceiver system, which lead to errors in high transition-rate data-patterns. Finally, the pattern-related failure is made more serious by the non-uniform nature of application data. We illustrate how these circumstances compound the *hot-spotting* effects; these will occur for any standard block code system.

To address this last issue, we applied a scrambler as a form of data-whitener and were able to successfully illustrate that its use removed the *hot-spotting* in the data-space. We conjecture that such a combination of 8B/10B block codec and a scrambler, while not improving the underlying loss-rate, can restore the uniformity of error which may be expected by higher level network layers, as well as restoring uniformity to the occurrence of data errors among data packets.

The IEEE 802.3z specification defines a robust network; at this layer, obeying the specification, engineers will not see the issues we have documented here. We consider the future of optical networks will implicitly alter the environment for those working at the packet layer through to the application layer. Developers of future optical networks should be aware that the behaviour of future physical and data-link layers may not be the same as those now deployed.

We have shown that naïve layering, the evolution of protocol layers beyond the scope of the original specification, together with the inadvertent loss of information between layers, can lead to unexpected errors as optical networks operate at higher data-rates with increasing complexity.

Acknowledgements

We thank Bradley Booth, Jon Crowcroft, David G. Cunningham, Eric Jacobsen, Peter Kirkpatrick, Tao Lin, Barry O'Mahony, Ralphe Neill, Ian Pratt, Adrian P. Stephens, and Kevin Williams.

REFERENCES

[1] W. Simpson, "PPP over SONET/SDH," IETF, RFC 1619, May 1994.
[2] A. Malis and W. Simpson, "PPP over SONET/SDH," IETF, RFC 2615, June 1999.
[3] ANSI, "T1.105-1988, Synchronous Optical Network (SONET) —Digital Hierarchy: Optical Interface Rates and Formats Specification," 1988.
[4] D. L. Tennenhouse, "Layered multiplexing considered harmful," in *Protocols for High-Speed Networks*. North Holland, Amsterdam, May 1989.
[5] IEEE, "IEEE 802.3z —Gigabit Ethernet," 1998, standard.
[6] J. S. Turner, "Terabit burst switching," *J. High Speed Netw.*, vol. 8, no. 1, pp. 3–16, 1999.
[7] P. Gambini *et al.*, "Transparent optical packet switching: network architecture and demonstrators in the KEOPS project," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 7, pp. 1245–1259, Sept. 1998.
[8] D. McAuley, "Optical Local Area Network," in *Computer Systems: Theory, Technology and Applications*, A. Herbert and K. Spärck-Jones, Eds. Springer-Verlag, Feb 2003.
[9] B. A. Small *et al.*, "Demonstration of a Complete 12-Port Terabit Capacity Optical Packet Switching Fabric," in *Proceedings of OFC-2005*, Anaheim, CA, Mar. 2005.
[10] I. H. White *et al.*, "Optical Local Area Networking using CWDM," in *SPIE ITCom 2003*, Orlando, FL, Sept. 2003.
[11] L. B. James *et al.*, "Wavelength Striped Semi-synchronous Optical Local Area Networks," in *London Communications Symposium (LCS 2003)*, Sept. 2003.
[12] IEEE, "IEEE 802.3ah —Ethernet in the First Mile," 2004, standard.
[13] A. Markopoulou *et al.*, "Characterization of failures in an IP backbone," in *Proceedings of IEEE INFOCOMM 2004*, Hong Kong, Mar. 2004, Sprint ATL Research Report.
[14] A. X. Widmer and P. A. Franaszek, "A DC-Balanced, Partitioned-Block, 8B/10B Transmission Code," *IBM Journal of Research and Development*, vol. 27, no. 5, pp. 440–451, Sept. 1983.
[15] The Fibre Channel Association, *Fibre Channel Storage Area Networks*. Eagle Rock, VA: LLH Technology Publishing, 2001.
[16] IEEE, "IEEE 1394b —High-Performance Serial Bus," 2002, standard.
[17] E. Solari and B. Congdon, *The Complete PCIExpress Reference*. Hillsboro, OR: Intel Press, 2003.
[18] "tcpfire," 2003, http://www.nprobe.org/tools/.
[19] A. W. Moore *et al*, "Architecture of a Network Monitor," in *Passive & Active Measurement Workshop 2003 (PAM2003)*, Apr. 2003.
[20] L. B. James *et al.*, "Beyond gigabit ethernet: Physical layer issues in future optical networks," in *Proceedings of London Communications Symposium*, 2004.
[21] L. B. James *et al.*, "Packet error rate and bit error rate non-deterministic relationship in optical network applications," in *Proceedings of OFC-2005*, Anaheim, CA, Mar. 2005.
[22] IEEE, "IEEE 802.3ae —10 Gb/s Ethernet," 2002, standard.
[23] IEEE, "IEEE 802.15.4 — Wireless Personal Area Network," 2003, standard.
[24] R. Jain, "Error Characteristics of Fiber Distributed Data Interface (FDDI)," *IEEE Transactions on Communications*, vol. 38, no. 8, pp. 1244–1252, 1990.
[25] L. B. James, A. W. Moore, and M. Glick, "Structured Errors in Optical Gigabit Ethernet," in *Passive and Active Measurement Workshop (PAM 2004)*, Apr. 2004.
[26] J. Stone, M. Greenwald, C. Partridge, and J. Hughes, "Performance of Checksums and CRCs over Real Data," in *Proceedings of ACM SIGCOMM 2000*, Stockholm, Sweden, Aug. 2000.
[27] J. Stone and C. Partridge, "When the CRC and TCP checksum disagree," in *Proceedings of ACM SIGCOMM 2000*. ACM Press, Aug. 2000.