2018-01-01

Detecting Contaminated Fiber Connectors Using Sfp Optical Power Data

Christopher A. Mendoza
University of Texas at El Paso, camendoza7@miners.utep.edu

DETECTING CONTAMINATED FIBER CONNECTORS USING SFP OPTICAL POWER DATA

CHRISTOPHER MENDOZA

Master's Program in Computer Engineering

APPROVED:

Michael P. McGarry, Chair, Ph.D.

Rodrigo Romero, Ph.D.

Eric Smith, Ph.D.

Charles Ambler, Ph.D.
Dean of the Graduate School

to my

Parents

with love

DETECTING CONTAMINATED FIBER CONNECTORS USING SFP OPTICAL POWER DATA

by

CHRISTOPHER MENDOZA, BSEE

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at El Paso

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE

Department of Electrical and Computer Engineering

THE UNIVERSITY OF TEXAS AT EL PASO

December 2018

Acknowledgments

I would like to acknowledge Dr. Michael P. McGarry for all his guidance and wisdom while completing this work. He has helped me grow as a researcher, academic and person. I would also like to thank both my parents and Nicole for supporting me and always assuring me that I would succeed. Finally thank you to the University of Texas at El Paso, the United States Army and the National Science Foundation for providing me with resources to complete this thesis.

Abstract

Fiber optic technology is an important part of communication networks enabling high-bandwidth transmissions over long and short distances. They do have their fair share of problems though, contamination being the biggest culprit. Contamination of fiber optic connectors can lead to serious performance degradation or even loss of signal. Detecting contaminated fiber connectors can take weeks or even months using traditional practices. There are standard cleanliness practices when dealing with optical connectors but still the problem seems to persist. This work presents an inequality to solve the detection portion of this problem. The proposed inequality uses power readings from the Small Form-Factor Pluggables' (SFPs) Digital Optical Monitoring (DOM) capabilities to detect if the contamination is affecting the optical signal. The inequality proposed also takes into account the tolerance range of the optical power readings, the suggested tolerance is $\pm$ 3 dB but this work shows that in practice it is much closer to $\pm$ 1 dB. The inequality is used to detect contaminated connectors in an experiment where power samples are collected over a day and is able to detect them with no false positives. A top-down approach is also taken to detect contaminated fiber connectors using higher layer event counters such as TCP retransmissions. After several trials using the top-down method, the results are inconclusive. Further work is needed to detect contaminated connectors using this method.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Communication networks are vital to how humans communicate, it is the foundation of the Internet. Reliable communication is necessary to help society progress. All types of networks rely on optical networks, not necessarily in their local network but there are many used for long-haul transmissions to connect to the Internet. The motivation for this work is to help alleviate some of the burdens of managing optical networks. Managing large scale optical networks can prove to be difficult which is why there is a trend in ongoing research to automate network management [2]. The steps towards automation are heavily influenced by machine learning and statistical methods to help network managers detect faults and correct for them in their networks. There are other pursuits to make network management easier as well, Software Defined Networking (SDN) is a new technology to help network managers configure and monitor their networks with tools like OpenFlow [3] [4]. SDN uses controllers to configure and monitor the network by communicating with switches that are OpenFlow (or similar protocol) enabled. SDN is based on the premise of splitting up the network into a data plane and control plane, the data plane is where all the user traffic goes through and the control plane handles all the control traffic sent to and from the controllers as shown in Figure 1.1.

The network manager is responsible for all the users and data that runs through that network. This can prove to be a daunting task as there can be many users using many devices all using the network at the same time. This is why network management has proven to be difficult, there are so many factors that could affect networks of all sizes and the network manager has to keep a watchful eye out for anything that could pose a threat to the integrity of the network. This includes trying to prevent issues before they

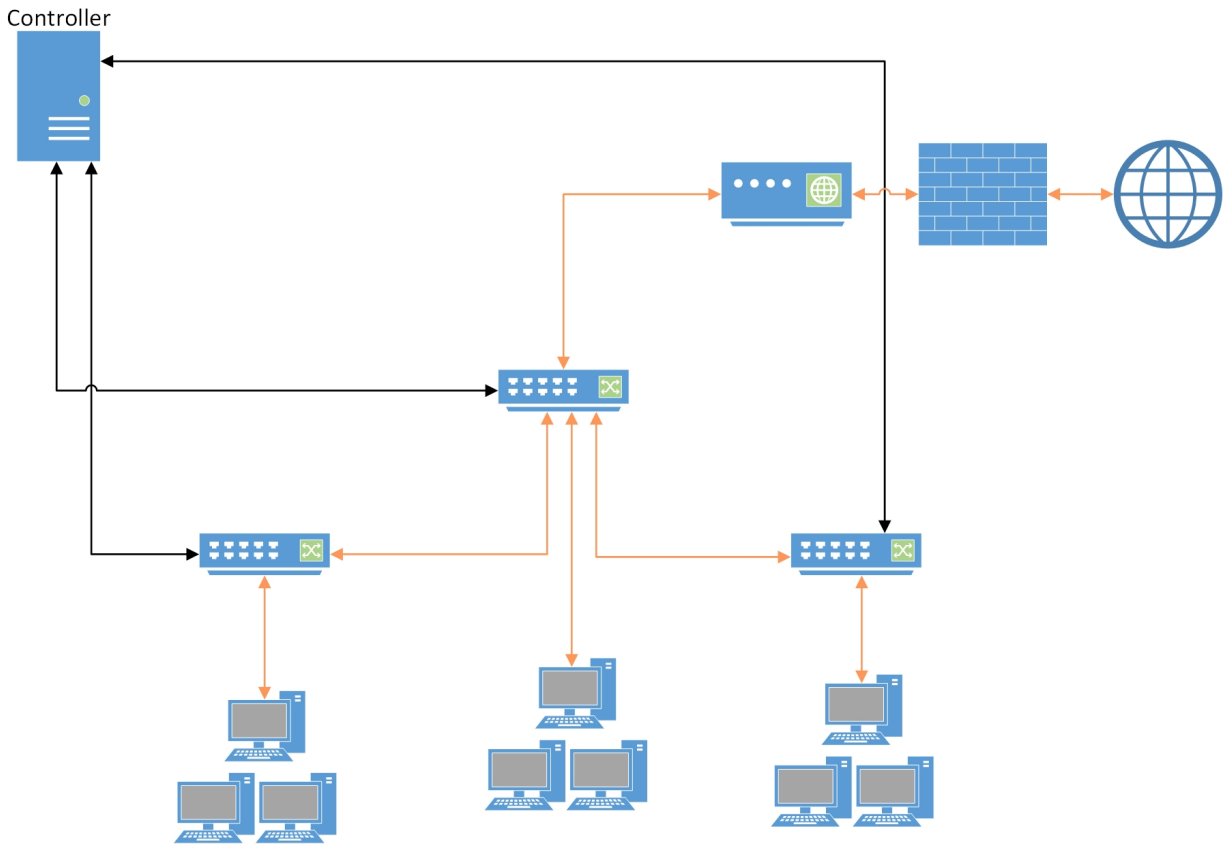


Figure 1.1: Small SDN network

happen, which tends to be difficult in a system with many random variables. As network size increases it becomes more difficult to manage. The classical FCAPS model, as shown in Figure 1.2 describes what a network manager's responsibilities are: fault detection, configuration, accounting/billing, performance assurance and security of the network.

Each one of these responsibilities is a task on their own but this work will look at the fault detection portion of the FCAPS model. There are two types of failures, hard-failures, and soft-failures. Hard-hard failures are when the component or service in question completely fails, such as a broken link. Soft-failures are when the component or service does not see complete failure but a reduction in performance or stability. Table 1.1 shows examples of each type of failure.

Contaminated fiber connectors can cause both types of failures, both soft and hard.

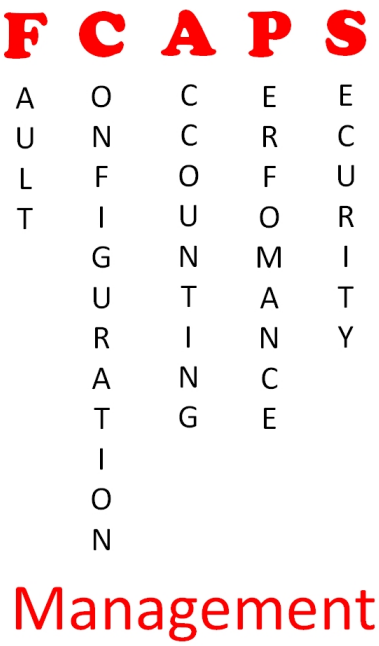


Figure 1.2: FCAPS model for network management

This has proven to be a frequent problem as most major network service providers estimate that over 70% of optical network troubleshooting is due to contaminated connectors [5]. Contamination of the optical connectors causes signal degradation which can lead to more bit errors or even complete link failure. On a large network, it is unfeasible to expend enough manpower to inspect the connectors by hand. Even on a smaller network, it is better to allocate resources elsewhere if possible. To try to detect this problem and allow the manager to prevent any possible failure is what this work aims to accomplish. The

Table 1.1: Soft vs Hard Failures

Soft-Failures	**Hard-Failures**
Increased Bit Error Rate (BER)	Cut wire
TCP misconfiguration	Loss of Signal (LoS)
Slow throughput	Server down

physical nature of this work means there are limitations of the resources available. There was no access to a large network to validate the method due to the unwillingness of any network manager to accept.

This work proposes an inequality to detect contamination that affects optical power. The inequality allows for detection without any added hardware. Experiments were conducted to validate this inequality by collecting optical power samples from the Small Form-Factor Pluggables' (SFPs) Digital Optical Monitoring (DOM) capabilities. The inequality is heavily reliant on the tolerance range of the SFP power measurements thus experiments were conducted to determine if the measurements from the SFP are accurate enough to detect contaminated optical connectors with certainty. This work also investigated using a top-down approach to detect symptoms of contaminated optical connectors higher in the data stack. This involved collecting many non-physical layer statistics, a list of all statistics collected can be found in Appendix A. The statistics with the most variance, TCP retransmissions and TCP delayed acknowledgments, were chosen as inputs for a K-Nearest Neighbors (KNN) and decision tree models to label new optical power samples as either clean or contaminated.

1.1 Outline of this Thesis

Chapter 2 describes related topics on fiber-optic communications, with information about gigabit Ethernet optical standards, transmitter and receiver technology, cabling information, Small Form-Factor Pluggables (SFPs), modulation techniques, typical loss measurements and more. Information about data analysis which includes machine learning, data mining, and general techniques when analyzing data is also included. Chapter 2 covers related literature about contamination of optical connectors and fault/anomaly detection.

Chapter 3 describes the experiments conducted for this research and what they aim to accomplish. This includes information about the topology of the test network, the data collected, the method to collect the data and all the logistics of how the experiment was

conducted.

Chapter 4 describes the results of the experiments described in Chapter 3. A formulation of an inequality to detect contaminated optical connectors, which is the main objective of this research, is discussed. The inequality is then verified on the data collected from the experiments. Chapter 4 also discusses the true tolerance range of SFP power measurements. Finally, Chapter 4 discusses the top-down approach to detecting contaminated fiber connector symptoms using higher layer statistics.

Chapter 5 contains the conclusions of this work and discusses future work.

Chapter 2

Fiber Optic Communications

2.1 Single-Mode Fiber (SMF) vs Multi-Mode Fiber (MMF)

Single-mode and multi-mode fiber cables allow for high-speed data transfers by being the medium through which optical transmitters and receivers, i.e lasers and photodiodes, communicate. The main difference between the single-mode and multi-mode fiber is the transmitter type and the diameter of the core.

Single-mode fiber has a core diameter of 9 µm as shown in Figure 2.1 and only has one mode/path of light as shown in Figure 2.2. Single-mode fiber operates within the wavelengths of 1310 nm and 1550 nm which gives less attenuation per meter. Due to these properties, single-mode fiber is ideal for long-distance data transmissions.

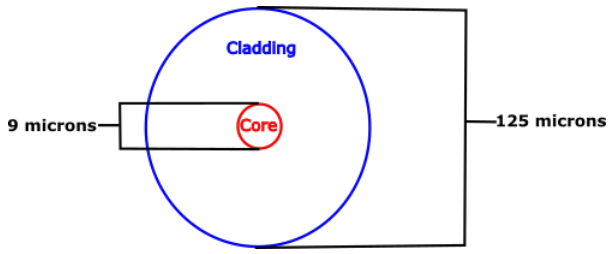


Figure 2.1: Core to cladding ratio of single-mode fiber cables

The diameter core of multi-mode fiber is usually 50 µm or 62.5 µm as shown in Figure 2.3 which has a much wider core diameter than it's single-mode counterpart. Having a much larger core diameter allows surface emitting lasers to couple nicely with these cables

Figure 2.2: Single-mode light path

allowing many more paths/modes of light as shown in Figure 2.4, hence the name multi-mode. Having a large core diameter and multiple modes of light causes higher attenuation per meter which is why multi-mode fiber is usually restricted to short-haul transmissions.

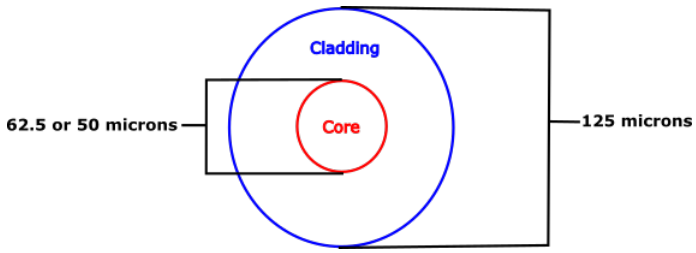


Figure 2.3: Core to cladding ratio of multi-mode fiber cables

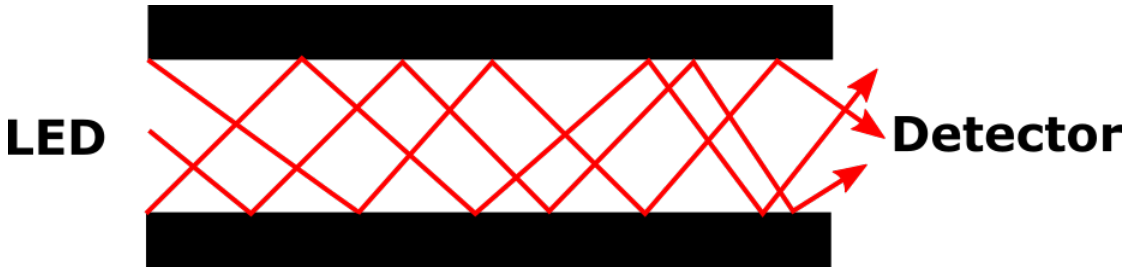
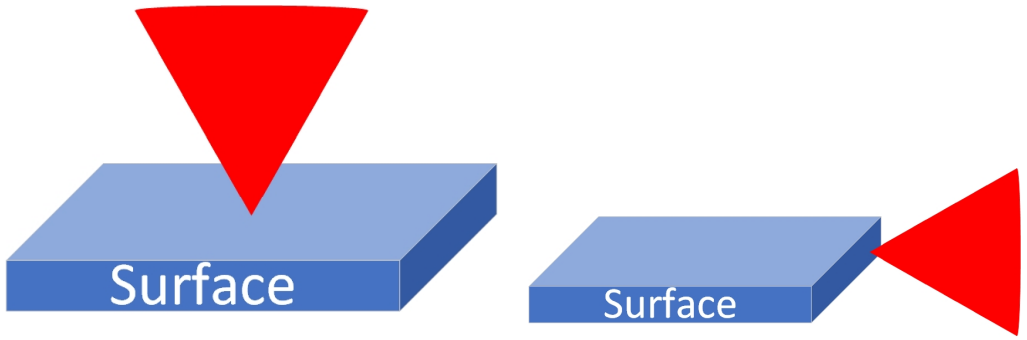


Figure 2.4: Multi-mode light paths

2.1.1 Transmitters and Receivers

There are two main types of lasers used for high bandwidth fiber optic communications, surface emitting lasers and edge-emitting lasers as shown in Figure 2.5. Surface emitting lasers generate light that leaves the device surface at a perpendicular angle, as where edge



a) Surface Emitting Laser b) Edge Emitting Laser

Figure 2.5: Transmitters

emitting lasers generate light that leaves the device at an angle parallel to the surface. The most commonly used type of surface emitting lasers are Vertical-Cavity Surface-Emitting Lasers (VCSELs)[6]. VCSELs are good for high bandwidth short-haul transmissions due to the output optical signal having a short wavelength generally, 850 nm [7]. VCSELs are made using 2 different Bragg reflectors with an active region in the middle, the commonly used materials for the reflectors are GaAs and AlGaAs. The most commonly used type of edge emitting lasers is Fabry-Pérot lasers (FP Lasers). FP lasers are generally used for long-haul transmissions due to being able to produce high power long wavelength optical signals, which are essential for communication over long distances. This is achieved by using two highly reflective and slightly transmitting parallel mirrors, the concept stems from the Fabry-Pérot resonator but is applied to lasers for optical communication, thus receiving the name Fabry-Pérot lasers [8]. Those are the two main types of transmitters used for gigabit optical networking but there are other transmitter technologies used such as Distributed FeedBack (DFB) lasers/Directly Modulated Lasers (DMLs) and Electro-absorption Modulated Lasers (EMLs).

Photodiodes are used for fiber optic communication receivers. However, depending on the wavelength of the optical signal of the transmitter, the photodiode must be made of different materials. The materials affect the wavelength at which the photodiode is the most sensitive. The materials for the single-mode and multi-mode receivers are strategically

selected differently to have higher sensitivity at the appropriate transmitter wavelength. For multi-mode wavelength (850 nm) a simple silicon photodiode is used. For single-mode wavelengths (1310/1510 nm) more sensitive devices need to be used, thus the common receiver to use is a InGaAs based photodiode, the reason that this material is used over Ge is that it can provide a better signal quality, i.e produces less noise. There are also avalanche photodiodes (APDs) that are commonly used for long-haul communications. APDs are used because they can reach higher sensitivity due to their built-in gain [9]. However, since the whole signal gets amplified it is essential that it has a high signal to noise ratio.

2.2 Small Form-Factor Pluggables (SFPs)

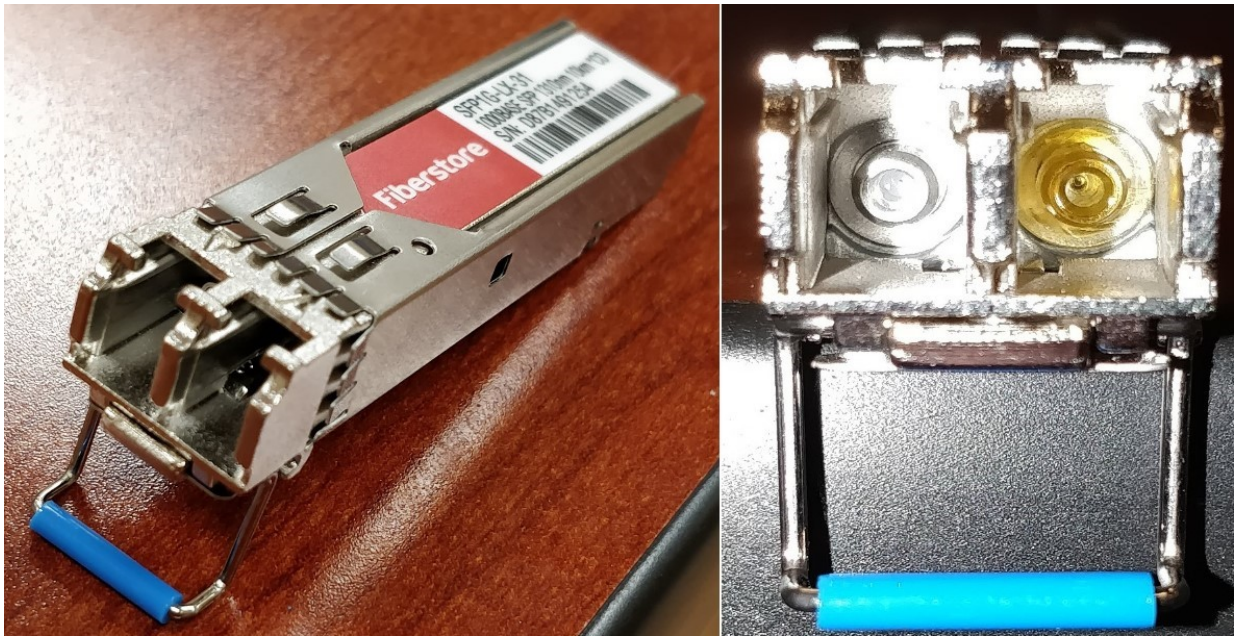


Figure 2.6: Small Form-Factor Pluggable

Small form-factor pluggables, as shown in Figure 2.6, are critical to transfer data in optical networks due to their plug-and-play nature and reliability to transfer data at high speeds. SFPs are modular devices that plug into other networking equipment such as

switches and optical taps. SFPs for optical communications contain the appropriate transmitters and receivers according to certain standards. Switch manufacturers will produce switches with numerous SFP ports rather than hardwired transceivers due to the modularity and flexibility it adds to their devices. SFPs come in many different forms which gives the user control of their network, letting them choose the bitrate, connector type, and standards the SFP adheres to. SFP builds are regulated by three standards:

- SFF-8472 [10]
- SFP Multi-Source Agreement (MSA) [11]
- IEEE 802.3 Ethernet Standard [12]

The SFF-8472 was created by the Small Form-Factor Committee, now known as the Storage Networking Industry Association (SNIA), was made for the purpose of standardizing what information the SFPs should collect, how the data should be represented and giving tolerance ranges for certain statistics. The SFP MSA standardizes the physical aspects of the SFPs such as the dimensions of the unit, timing requirements of the I/O and the pin layout on the PCB. The IEEE standards are in the 802.3 Ethernet document [12], this describes the physical requirements of the SFP transmitter and receiver in the PMD portion of each particular standard.

2.3 Cables and Connectors

There are many different cable types and connectors available. A few notable examples are the Subscriber Connector (SC), the Lucent Connector (LC), the Straight Tip (ST) and the Ferrule Connector (FC) as shown in Figure 2.7. Of the list, only the LC connector has a 1.25 mm ferrule as shown in Figure 2.8. The rest of the connections all have 2.5 mm ferrules.

Each of these connectors differs in cost and reliability so the ideal connector depends on the use case. These connectors can all have different polishes, which is the way that

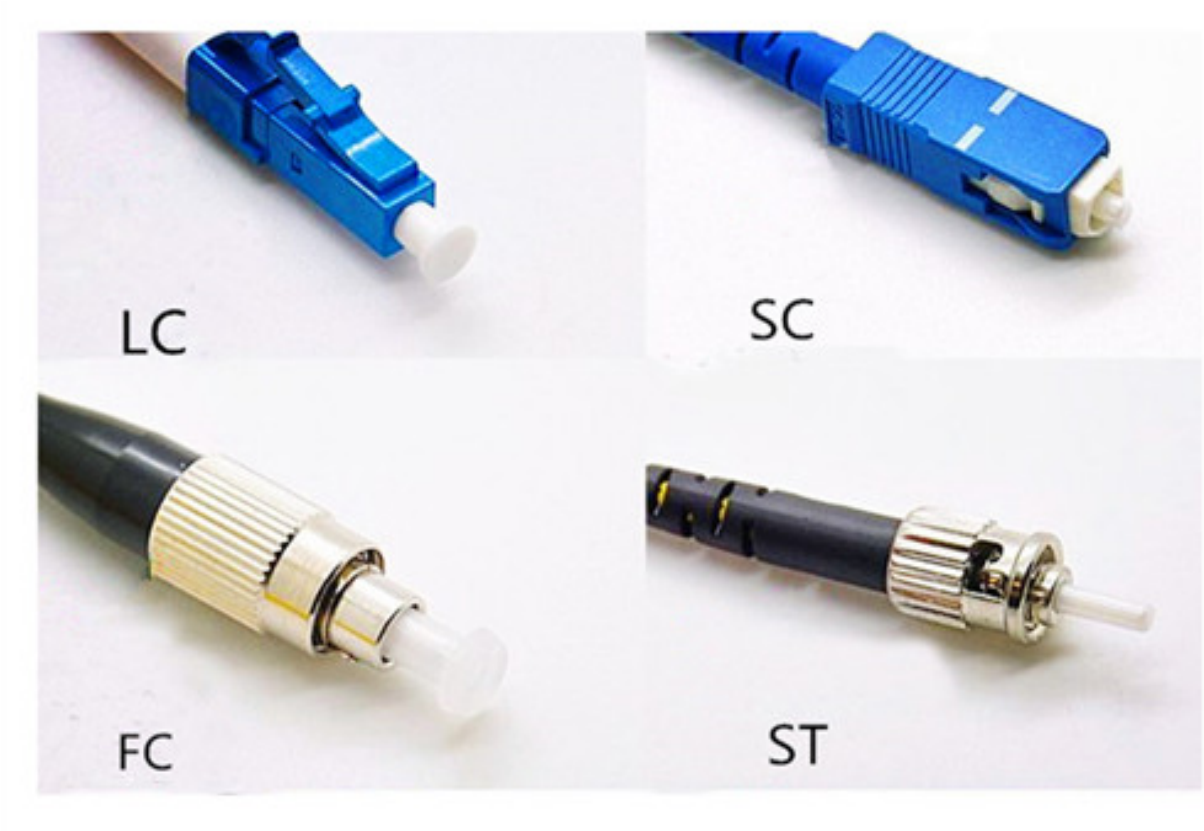
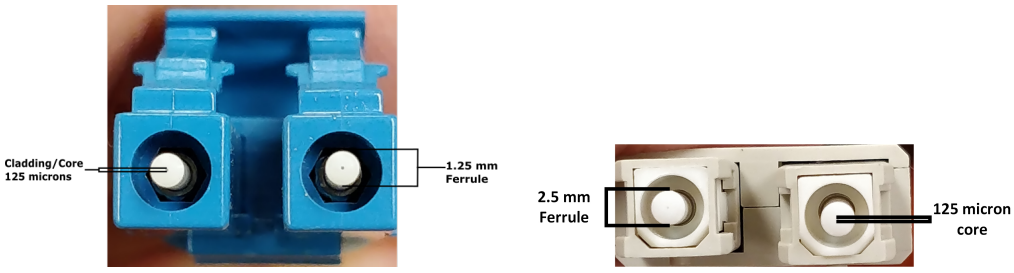


Figure 2.7: Fiber-optic connectors [1]



a) Lucent Connector

b) Subscriber Connector

Figure 2.8: LC and SC Connectors

the ferrule is shaped. There are 3 types of polishes, Physical Contact (PC), Ultra Physical Contact (UPC) and Angled Physical Contact (APC). Having a PC polish means that only the cores will come into contact, UPC will also only have the cores contact but with greater precision and the APC will angle the connector. This is done to reduce return loss caused

by the connectors, so depending on the requirements of the channel it may be ideal to choose a polish that will minimize return loss.

SFPs contain sub-assemblies for both the transmitter and receiver. There are three different types of sub-assemblies, Transmitter Optical Sub-Assembly (TOSA), Receiver Optical Sub-Assembly (ROSA) and Bi-directional Optical Sub-Assembly (BOSA). The TOSA and ROSA contain the hardware pertinent to the transmitter or receiver respectively or in the case of the BOSA both. The TOSA will contain the laser diode, any type of lenses necessary, along with any other add-ons the manufacturer wants to add, such as photodiode monitors. The ROSA is similar, it contains the photodiode, along with lenses and add-ons. The BOSA is a combination of both, by using Wave Division Multiplexing (WDM) on one fiber. The BOSA is not popular as it is much more cost effective to have the TOSA and ROSA separate and use two separate fibers.

2.4 Modulation

Optical transmitters in SFPs use Pulse Amplitude Modulation with two levels (PAM-2), meaning that each unique bit will be represented by one of two power levels as shown in Figure 2.9. The received optical signal is shown in Figure 2.10, where there is a time shift due (τ) to propagation delay and attenuation (δ). This is the signal the photodiode will detect and convert into an amplified (α) electrical signal (V) to be used by the serializer/deserializer, as shown in Figure 2.11.

Modulation is accomplished by using a driver circuit to modulate the laser, as shown in Figure 2.12. The driver uses the inputs produced by the Physical Medium Attachment (PMA) sublayer which are the serialized bits of the 10-bit codewords produced by the Physical Coding Sublayer (PCS). The 8b/10b encoding is essential for optimal performance of the transmitter, receiver and cables.

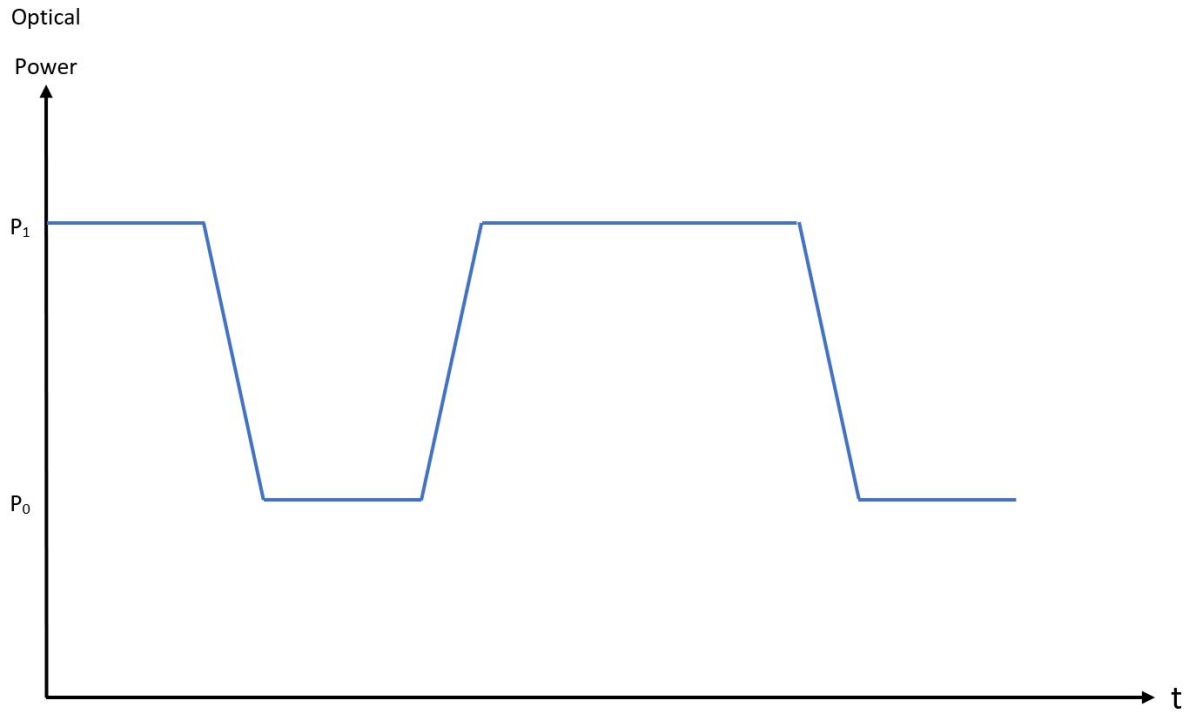


Figure 2.9: Transmitted optical signal using PAM-2

2.5 Loss Measurements

The two categories of loss in fiber optic communications are insertion loss and return loss.

2.5.1 Insertion Loss (IL)

Insertion loss is the ratio of power transmitted over the power received. This is usually measured in decibels, thus the equation for insertion loss is given by:

$$IL = 10 \log_{10} \left(\frac{Tx\ Power}{Rx\ Power} \right) \tag{2.1}$$

This measurement is useful as it will not only include the natural power attenuation from distance and passive components but also the power loss due to contamination.

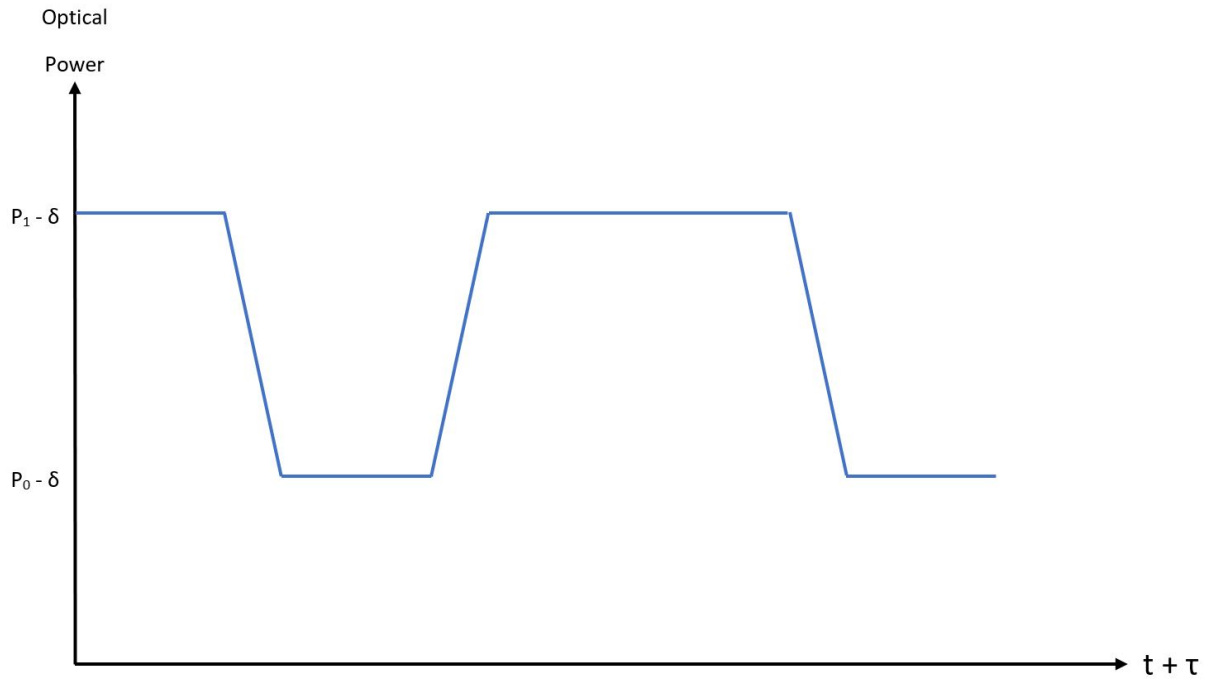


Figure 2.10: Receiver optical signal using PAM-2

2.5.2 Return Loss (RL)

Return loss is the ratio of power incident to the power reflected back at a surface, where the surface is fiber optic connector end-faces. This is also usually measured in decibels, thus the equation is given by:

$$RL = 10\log_{10}\left(\frac{Tx\ Power}{Reflected\ Power}\right) \tag{2.2}$$

The power reflected back can be computed by using Fresnel's laws of reflections. Where P is the reflection coefficient from the case where the electric field is parallel to the surface and S is the reflection coefficient from the case where the electric field is perpendicular to the surface. Where P and S are given by:

$$P = \frac{-n_2^2 cos\theta_i + n_1\sqrt{(n_2^2 - n_1^2 sin^2\theta_i)}}{n_2^2 cos\theta_i + n_1\sqrt{(n_2^2 - n_1^2 sin^2\theta_i)}} \tag{2.3}$$

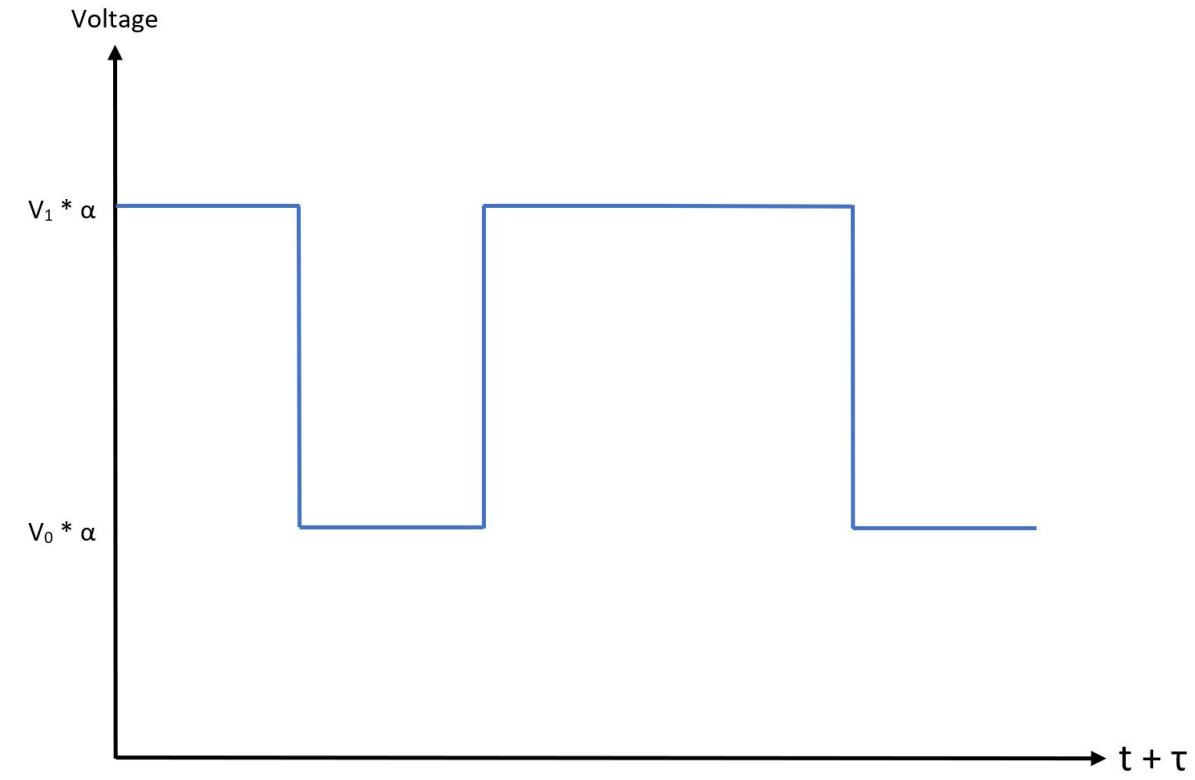


Figure 2.11: Electrical signal after being amplified/quantized

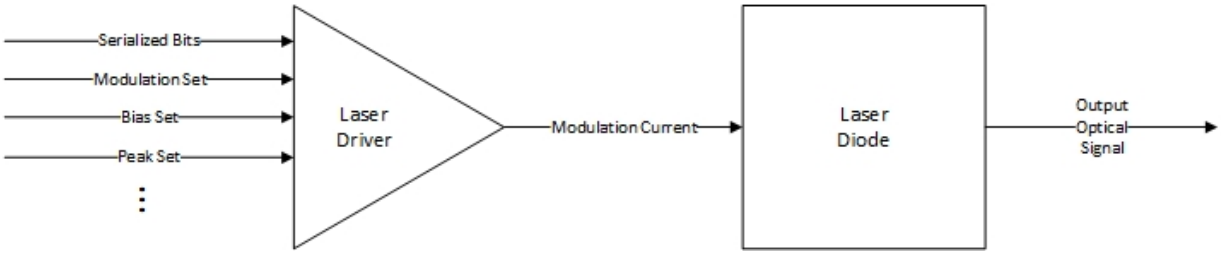


Figure 2.12: Driver taking multiple inputs to drive a laser diode

$$S = \frac{n_1 cos\theta_i - \sqrt{(n_2^2 - n_1^2 sin^2\theta_i)}}{n_1 cos\theta_i + \sqrt{(n_2^2 - n_1^2 sin^2\theta_i)}} \tag{2.4}$$

However in the special case where the angle of incidence is normal to the surface of

where the refractive index changes then Equation 2.3 can be simplified to,

$$S = \frac{n_1 - n_2}{n_1 + n_2} \tag{2.5}$$

Where the power reflected back is the square of S, if there is no parallel polarization.

$$Power\ Reflected = \left(\frac{n_1 - n_2}{n_1 + n_2}\right)^2 \tag{2.6}$$

$P = 0$ can be achieved by using Brewster's angle which is given by:

$$\theta_B = tan^{-1}\frac{n_2}{n_1} \tag{2.7}$$

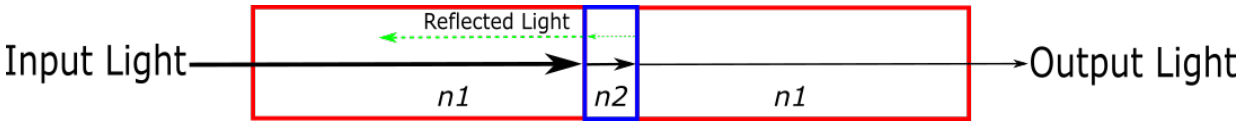


Figure 2.13: Return loss caused when light travels through media with different refractive indexes

This is a useful measurement when looking at the effect of contamination on optical performance due to the contaminants having a refractive index that will reflect some of the power back and cause power loss. Return Loss and Optical Return Loss (ORL) are interchangeable terms.

2.6 Optical Gigabit Ethernet Standards

Gigabit optical networking has two common standards which are 1000BASE-SX and 1000BASE-LX, which are both standards that are described in the IEEE 802.3 Ethernet standard [12]. Both of these standards are derivatives of the 1000BASE-X standard which is also described in the Ethernet 802.3 standard. The functionality of the devices that adhere to the aforementioned standards can be separated into 3 different sections, the Physical Coding Sublayer (PCS), the Physical Medium Attachment (PMA) and the

Physical Medium Dependent (PMD). Figure 2.14 shows the extension of the physical layer of the typical 5-layer OSI model to show the 3 sublayers.

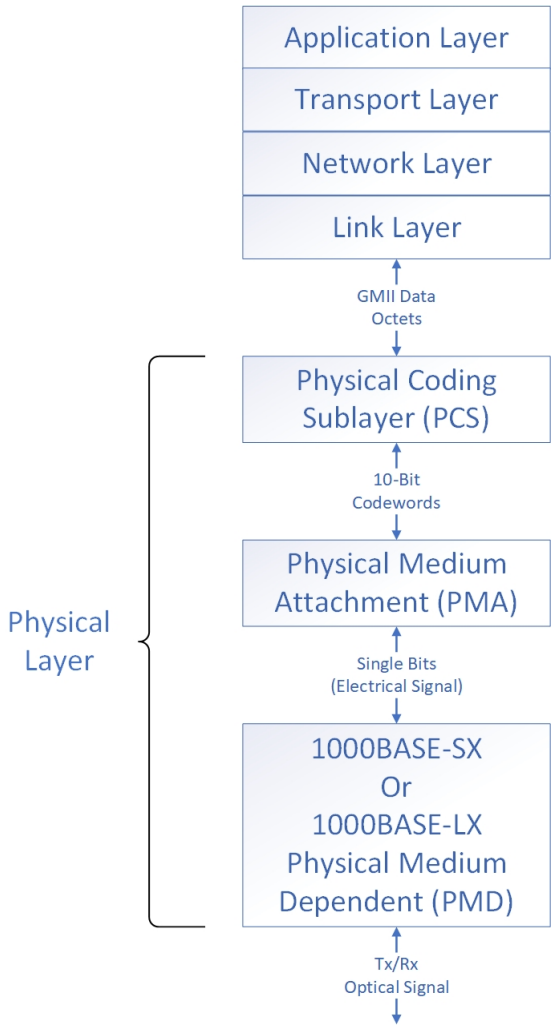


Figure 2.14: Extended OSI model to show sublayers

Both the PCS and PMA are the same for 1000BASE-SX and 1000BASE-LX standards which are identical to that of the 1000BASE-X specifications, however they do differ in PMD specifications due to having different requirements in the 802.3 standard as shown in Table 2.1. The average launch power is the average power of the optical signal produced by the transmitter. The average receive power is the average power of the optical signal detected by the photodiode. The receiver sensitivity is the minimum amount of power that

Table 2.1: 1000BASE-SX vs 1000BASE-LX PMD Characteristics, from Ethernet Standards

	1000BASE-SX	**1000BASE-LX**	**Units**
Transmitter Wavelength	770 to 860	1270 to 1355	nm
Range (Max)	550	5000	m
Average Launch Power (Max)	0	-3	dBm
Average Launch Power (Min)	-9.5	-11.5 to -11[a]	dBm
Average Receive Power (Max)	0	-3	dBm
Receiver Sensitivity	-17	-19	dBm
Stressed Receiver Sensitivity	-12.5 to -13.5[a]	-14.4	dBm
Return Loss (Min)	12	12	dB
RMS Spectral Width (max)	0.85	4	nm

[a] Depending on the type of cable and wavelength used.

needs to be detected by the receiver to function correctly. Similarly the stressed receiver sensitivity is the power that is needed for the receiver to operate correctly under heavy loads. The spectral width is the span of the spectrum of light that is emitted from the transmitter.

2.6.1 Physical Medium Dependent (PMD)

The PMD portion of the physical layer, when using optical connectors, is responsible for interfacing optical signals with an electrical interface. The PMD will turn the stream of bits represented by an electrical signal and use a driver to produce the appropriate optical signal through the laser transmitter. There are different types of transmitter technologies that can be used depending on the standard. The two main types are surface emitting lasers and edge emitting lasers. The PMD will also produce an electrical signal based on received optical signal through use of a photodiode. Generally this is put through a pre-

amplifier as well as a quantizer to create a stream of bits that were received. Figure 2.15 shows an example of a 1-bit resolution quantizer, in the case of fiber optic communications the input would be the output from the photodiode. The quantizer will transform the input signal from the photodiode into a higher and lower power to represent the bits which will be deserialized by the PMA into 10-bit codewords. This is usually all contained within the SFP as shown in Figure 2.16, which shows the flow of bits from the transceiver to Layer 2.

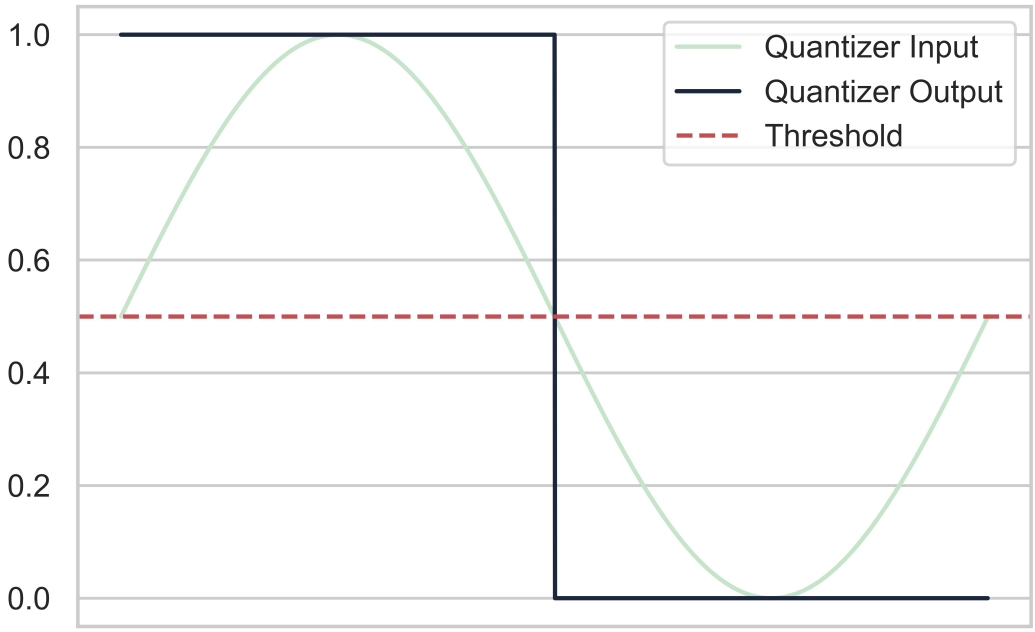


Figure 2.15: Quantizer example, where the input signal would be the electrical signal produced by the optical receiver i.e photodiode

2.6.2 Physical Medium Attachment (PMA)

The Physical Medium Attachment (PMA) is responsible for providing an interface between the Physical Medium Dependent (PMD) and the Physical Coding Sublayer (PCS). This is

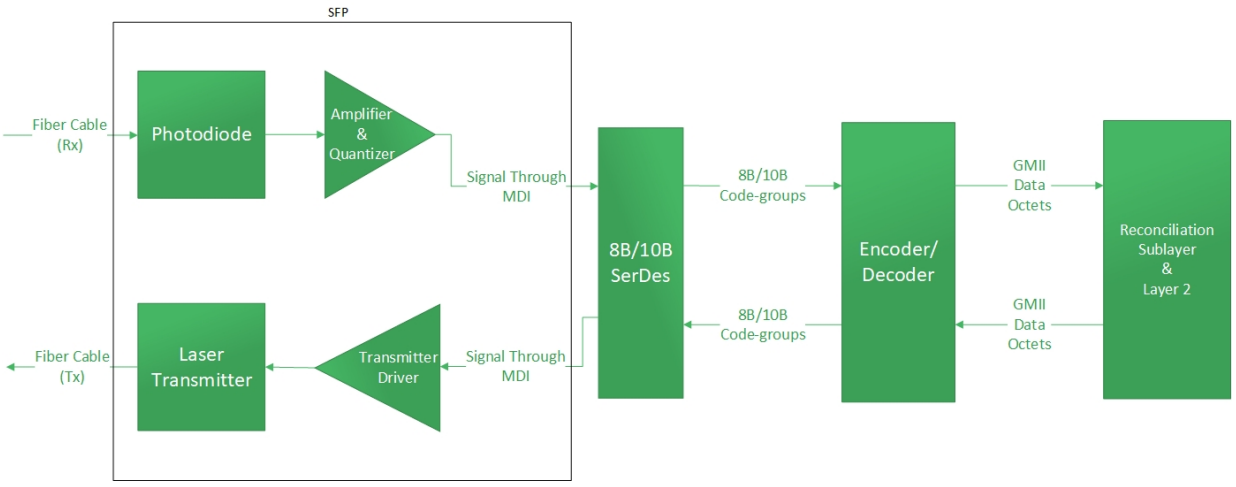


Figure 2.16: Flow of bits from optical signal to 8-bit bytes usable by Ethernet

to provide a layer of abstraction so that the PCS can interface with any PMD, which is accomplished by using a Media Dependent Interface (MDI) to exchange data between the PMA and the PMD. Thus the responsibilities that the PMA have are: serialization of the 10-bit characters received from the PCS, deserialization of the bits provided by the PMD into 10-bit characters, clock recovery from the 10-bit characters and data loopback. This is done using an 8b/10b SerDes IC along with protocol ICs and Programmable Logic Devices (PLDs) that allow for other statistics such as optical power levels through Digital Optical Monitoring (DOM) and Loss of Signal (LoS) to be reported.

2.6.3 Physical Coding Sublayer (PCS)

The PCS portion of the physical layer is responsible for interfacing the PMA with the Gigabit Media Independent Interface (GMII). It encodes GMII data octets into 8b/10b code groups to be used by the PMA and decodes 8b/10b code groups into GMII data octets to be used by layer 2. It is also in charge of handling collision detection and auto-negotiation.

8b/10b is a line coding scheme that will take 8-bit bytes from the Layer 2 and will encode them into 10-bit codewords. The purpose of doing this is to achieve a DC balance,

i.e send the same amount of 1's as 0's. It also helps with clock recovery by ensuring that there are enough transitions. It splits the 8-bits into 5-bit and 3-bit groups then appends an extra bit at the end of the corresponding group. The tables for encoding/decoding are shown in Tables 2.2 and 2.3 along with the special codewords shown in Table 2.4. The output is depending on the running disparity which is given by (number of 1s sent - number of 0s sent), which starts by default at -1.

Table 2.2: 5b to 6b Encoding

Code Group	**Input Bits**	**IF RD = -1**	**IF RD = 1**
D.00	00000	100111	011000
D.01	00001	011101	100010
D.02	00010	101101	010010
D.03	00011	110001	110001
D.04	00100	110101	001010
D.05	00101	101001	101001
D.06	00110	011001	011001
D.07	00111	111000	000111
D.08	01000	111001	000110
D.09	01001	100101	100101
D.10	01010	010101	010101
D.11	01011	110100	110100
D.12	01100	001101	001101
D.13	01101	101100	101100
D.14	01110	011100	011100
D.15	01111	010111	101000
D.16	10000	011011	100100
D.17	10001	100011	100011

Table 2.2: 5b to 6b Encoding

Code Group	**Input Bits**	**IF RD = -1**	**IF RD = 1**
D.18	10010	010011	010011
D.19	10011	110010	110010
D.20	10100	001011	001011
D.21	10101	101010	101010
D.22	10110	011010	011010
D.23	10111	111010	000101
D.24	11000	110011	001100
D.25	11001	100110	100110
D.26	11010	010110	010110
D.27	11011	110110	001001
D.28	11100	001110	001110
D.29	11101	101110	010001
D.30	11110	011110	100001
D.31	11111	101011	010100
K.28	11100	001111	110000

Table 2.3: 3b to 4b Encoding

Code Group	**Input Bits**	**IF RD = -1**	**IF RD = 1**
D.x.0	000	0100	1011
D.x.1	001	1001	1001
D.x.2	010	0101	0101
D.x.3	011	0011	1100

Table 2.3: 3b to 4b Encoding

Code Group	**Input Bits**	**IF RD = -1**	**IF RD = 1**
D.x.4	100	0010	1101
D.x.5	101	1010	1010
D.x.6	110	0110	0110
D.x.P7	111	0001	1110
D.x.A7	111	1000	0111
K.x.0	000	0100	1011
K.x.1	001	1001	0110
K.x.2	010	0101	1010
K.x.3	011	0011	1100
K.x.4	100	0010	1101
K.x.5	101	1010	0101
K.x.6	110	0110	1001
K.x.7	111	1000	0111

Table 2.4: Special 8b/10b Codewords

Code Group	**Input Bits**	**IF RD = -1**	**IF RD = 1**
K.28.0	000 11100	001111 0100	110000 1011
K.28.1	001 11100	001111 1001	110000 0110
K.28.2	010 11100	001111 0101	110000 1010
K.28.3	011 11100	001111 0011	110000 1100
K.28.4	100 11100	001111 0010	110000 1101
K.28.5	101 11100	001111 1010	110000 0101

Table 2.4: Special 8b/10b Codewords

Code Group	Input Bits	IF RD = -1	IF RD = 1
K.28.6	110 11100	001111 0110	110000 1001
K.28.7	111 11100	001111 1000	110000 0111
K.23.7	111 10111	111010 1000	000101 0111
K.27.7	111 11011	110110 1000	001001 0111
K.29.7	111 11101	101110 1000	010001 0111
K.30.7	111 11110	011110 1000	100001 0111

2.6.4 1000BASE-SX

SX means that it uses a short wavelength laser, between 770 nm to 860 nm, to transmit optical signals across the physical medium, i.e fiber cables. The most popular laser technology for short-wavelength/multi-mode fiber is surface-emitting lasers such as Vertical-Cavity Surface-Emitting Lasers (VCSELs), however other laser technologies can be used if they meet the specifications in the 802.3 standard. The reason for using this technology is that it has a small enough emission area to couple well with multi-mode fiber cables which have a core diameter of either 50 microns or 62.5 microns. This technology is used for data transmission along shorter distances, rated for 550 meters using the correct cables. The receiver is a photodiode that can detect the appropriate wavelength, in this case being 770 to 860 nm.

2.6.5 1000BASE-LX

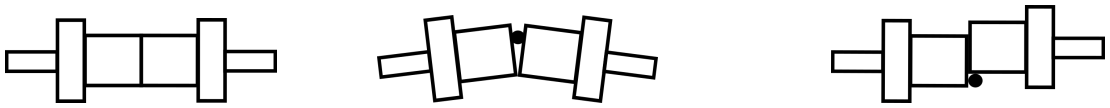
LX means that it uses a long wavelength laser, between 1270 nm and 1355 nm, to transmit optical signals across the physical medium. The most popular laser technology for long wavelength/single-mode fiber is edge-emitting lasers such as the Fabry-Pérot lasers (FP

Lasers). Since surface-emitting laser technologies have too large of an emission area to be coupled with the smaller 9 micron core diameter of single-mode fiber, edge-emitting technologies must be used to couple efficiently allowing for longer transmission distances. Longer wavelengths are ideal for transmitting longer distances, which is why 1000BASE-LX is rated for a transmission distance of 5 kilometers using the correct cables. The receiver is a photodiode that can detect the appropriate wavelength, in this case being 1270 to 1355 nm.

2.7 Contaminated Fiber Connectors

Contamination of fiber optic connectors can present issues in a network due by causing insertion loss, which includes return loss. The contaminants will generally have a different refractive index than the fiber causing some return loss. Contamination will also cause insertion loss by blocking light from coupling into the core of the fiber connector. Misalignment of connectors is also a culprit of optical signal degradation, which can be caused by either tilting or offsetting the connectors. Tilt is when the connectors do not form a 180 angle when mated. This can happen due to contamination residing in between two connectors causing them to not form a perfect flat connection. Offset is when they are not aligned in space and can also be caused by contamination residing between the bottom, top or sides of a connector. These scenarios are illustrated by Figure 2.17. All of these factors cause the signal quality to be degraded which can cause bit errors, which the following appropriate action Ethernet takes is to drop the frame. If there are many bit errors then the performance of the optical network will be severely affected and possibly cause complete link failure.

There was no literature that was directly related with the detection of contaminated fiber connectors in a network, however there has been work done in analyzing optical performance degradation due to contamination of the connectors, as well as anomaly detection within a network.



a) Good connection b) Misalignment due to tilt c) Misalignment due to offset

Figure 2.17: Connection Misalignment

2.7.1 Contamination's effect on optical performance

[13] presents an experimental study that provides insight into how fiber connector contamination and scratches affect optical signal performance. In their experiments Insertion loss (IL), optical return loss (ORL) and bit error rate (BER) were observed. These experiments used standard connector (SC) simplex single mode fiber at a rate of 10 Gbps and a variable attenuator. The procedure they followed was to clean, contaminate/scratch, then visually inspect and run performance tests. For carbon particle contamination, IL was increased by up to ten times, return loss was decreased by 2 to 3 times and the bit error rate was increased by 2 to 10 times. Particles have little to no effect if they are not located near the core and oil contamination has no effect on insertion loss due to it having a similar refractive index as the fiber material used. Oil has a large effect on return loss and carbon particle contamination has a large effect on the bit error rate.

[14] presented a slightly more extensive experimental study to analyze the effects of both incorrectly cleaved fiber ends and fiber connector contamination on optical signal performance. This study also used SC connectors. Incorrectly cleaved fiber ends can cause IL over 40 dB and ORL under 30 dB. The effect of fiber connector contamination are:

- Light blocking caused by contamination on the core can cause IL upwards of 7dB and can cause ORL to reduce depending on the refractive index of the contamination at 80% coverage.
- A 50 μm gap caused by contamination resulted in up to 1 dB of IL.
- Misalignment due to tilt caused by contamination resulted in IL upwards of 1.4 dB

at a 3 degree tilt.

- Misalignment due to offset caused by contamination resulted in IL upwards of 1.9 dB at a 3μm offset.

[15] developed cleanliness standards for single mode connectors. SC/FC connectors which are 2.5 mm in diameter and LC/MU connectors which are 1.25 mm in diameter. The most critical region was found to be the core area, which is the area within 25μm of the core. Contamination has the greatest effect on optical performance if located within this critical region. Each connector type was contaminated with Arizona dust and used a pass-fail system to test optical performance. The test subject had to fall within 3 times the Standard Deviation (SD) of the clean connector for both IL and ORL to pass. 2.5 mm ferrules were more resistant to particle movement due to mating/de-mating. 60% of the LC/MU connectors saw an increase of IL between 0.5 dB and 1.1 dB as a result of contamination and particle migration.

[16] showed that the primary source of contamination was due to the dust caps that the cables were shipped with. A suggestion was made to use a different material to help reduce the likelihood of contamination due to dust caps. Optical performance was tested to analyze the effects of the dust cap contamination. The results showed that the BER was 100 times larger and could even cause complete failure when the core of the fiber optic cable was blocked/contaminated.

[17] presented an experimental study of the effect of Arizona dust contamination on small form-factor pluggable (SFP) transceiver receptacles. Initially this study observed Optical Power, Pulse Mask Margin, Spectral Width and Optical Return Loss (ORL) however it was found that only ORL was being affected by the contamination and therefore that is the only parameter that was studied further. The contamination of the receptacles and single-mode fiber showed a common trend where the distance of the nearest contamination particle from the core correlated to ORL. The closer to the core the higher the difference in ORL was which could be as high as a 20-30 dB difference.

[18] analyzed the effects on short reach 10 Gb/s SFP+ transceivers which differ from other lower bandwidth transceivers because they have lensed receptacles rather than fiber stub interfaces. The lenses of the Optical Sub-Assemblies (OSAs) were contaminated using SiO_2 test dust, which was followed by 10 samples of the transmitted and received power. The results showed that the contamination had an extreme effect on 5 of the 10 samples and that the maximum difference between the Tx power and Rx power were 1.24 dBm and 1.15 dBm respectively. The results from the physical experiment were compared with simulated results using the ZEMAX software. The results of both the physical and simulated experiments were near identical, verifying that the contamination is having a negative effect on optical performance.

[5] study showed that there is a correlation between contamination and signal degradation in single-mode APC connectors. The experiments conducted involved 25 APC single-mode connectors that were mated and de-mated to analyze the effects of particle migration and contamination on optical connectors. It was found that if there is contamination found within the core there was significant RL reduction, an average of 14.2 dB. However, if there was only contamination outside the core area then there was little to no RL issues. In these experiments particle migration is most prevalent with particles < 5 microns. This study also showed that scratches have no effect on APC connectors even though scratches are known to cause a noticeable RL decrease in UPC single-mode connectors. It is also stated in this work that most major network service providers estimate that over 70% of optical network troubleshooting is due to contaminated connectors.

[19] conducted experiments to see the effect of contamination, particularly in aerospace applications, on single mode fiber. They conducted tests using different connectors such as butt connectors and several expanded beam connector types. They contaminated each type of connector with either water, potassium formate, Clearway3, Hyjet V, brake dust and a mixture of Hyjet V and brake dust. Results showed that there was little to no loss, $<$ 0.2 dB on clean butt connectors, and there was slightly more in the expanded beam types due to extra optics. The contamination degraded the performance differently depending on

the type of contaminant, where the worst case being brake dust on the butt connector with a loss of 83 dB. Insertion loss went up in all cases where a contaminant was introduced to the core area of the cables.

[20] conducted experiments where they used a cassette cleaning tape to clean connectors and placed them near metallic particles to demonstrate how electrostatic charge can be a catalyst for attracting contamination. Another source of contamination that was explored in this study the PVC dust caps that are used to cover the cables when not in use. The cables were covered with PVC dust caps that were contaminated with metallic particles and the results showed that the cables became contaminated over time, reaching peak contamination after 7 days. This study shows that not only dust can impact a network but any particle that obstructs the light path.

[21] used an analytic model to estimate the Return Loss (RL) on scratched optical fiber end faces based on size, location and relative reflectivity. The model produced used a new parameter introduced in this paper called relative reflectivity, which is described as the ratio between the reflectivity of a scratch compared to the reflectivity of an end-face with no defects. Using this particular model the RL induced by the scratch was accurately predicted within ± 1 dB.

Not only does contamination of fiber connectors degrade network performance but it can also damage the hardware itself. [22] conducted a study to find a correlation between damage to connectors and IL/RL, contamination location and power loss. If the IL and RL measurements do not meet the standards of > 0.5 dB and < 45 dB respectively, then it is likely that the connector will be damaged during high power transmission. The particle location also had a significant impact on the connector being damaged if the particles overlapped most of the core. It was found that if power loss exceeded 0.22 dB then the connector would likely be damaged in a high-power system. This study also showed that cleaning the connectors with both an air duster and optical connection cleaner was more effective than using just one method alone.

Table 2.5: Anomaly Detection

Statistical	**Machine Learning**	**Data Mining**
Descriptive	System call based sequence analysis	Clustering
Signature	Bayesian networks	Fuzzy logic
Anomaly score	Principle component analysis (PCA)	Association rule mining
Hotellings T^2 test	Neural networks	K-Nearest Neighbors (KNN)
Markov chains	Support Vector Machines (SVM)	

2.7.2 Fault/Anomaly Detection

[23] gives an overview of modern anomaly detection techniques. The techniques can be split up into 3 different sets: statistical, machine learning and data mining. Each set has pros and cons. Statistical techniques have the advantage of not needing any training data of anomalous behavior, it only needs to model the system under normal conditions. The cons are that network data is very difficult to model and is often non stationary. Machine learning can improve its modeling based on past results. The downside is that training data is required to train the models. Data mining techniques are good because they do not need training data and can often find patterns in data with no modeling. Data mining techniques do in turn produce high false positive rates. Table 2.5 shows each category with some example techniques listed in [23].

[24] takes a statistical approach to find network anomalies in an Internet Protocol (IP) network. Using a statistical signal processing approach they compare two consecutive time-windows to looks for changes in statistical features, specifically variance. The time series values of both windows are fed into an Auto-Regressive (AR) model which produces residuals for each window. The two residual windows' variances are compared to detect abrupt changes. This method was able to successfully detect file server errors, protocol errors, network access errors and runaway processes. However, this method is not resilient against sample loss, which is highly likely due to using SNMP to collect statistics.

[25] leverages Software Defined Networking (SDN) to try and detect network anomalies and show that past SDN based techniques will be more efficient when used in a small network environment. 4 different SDN anomaly detection techniques were tested, Threshold Random Walk with Credit Based Rate Limiting (TRW-CB), rate limiting, maximum entropy detector and NETAD. TRW-CB keeps track of how many new TCP connections are made which timeout or are reset. The more connections that are not acknowledged the more likely the host is compromised. Rate limiting has a "working set" of recent connections for each host. When the machine attempts to make a new connection if it is not in the working set it is put in a queue. The queue is served at a certain rate and if the queue becomes larger than a certain threshold then the host is declared as infected. Maximum entropy detectors work by estimating the distribution of the measured features when in a normal state. Then as new samples are recorded they are compared to the "normal" acting distribution, the similarity is computed using the Kullback-Leibler (KL) divergence measure. NETAD is a rule-based method to detect anomalous packets. "Non-interesting" traffic is removed and not analyzed. Each packet that is analyzed is given a score which is based on time and frequency of similar packets received. A packet with a high enough score is flagged as anomalous. The four techniques were tested on a home network, a home office network and an ISP network. The results show that these methods are much more effective in smaller networks with much higher detection rates with much lower false positives.

[26] uses 8 different machine learning algorithms to test for anomalous behavior in multi-core routers. The anomalous behavior would be invoked by Trojan viruses that cause core address spoofing, route looping and traffic diversion. The techniques used are K-Nearest Neighbors (KNN), Support Vector Machines (SVM), linear regression and decision trees, K-Means, farthest first, estimation maximization and hierarchical clustering. Features that are monitored: source core, destination core, packet transfer path, distance (number of hops), power range, execution time range, clock frequency and supply voltage. The accuracy of the supervised learning models performed well, achieving 90% or higher accuracy, while the unsupervised models did not.

[27] introduces a network anomaly detection framework, BasisDetect, that introduces an innovative way to process time series data collected in a network. They form a dictionary that breaks up each link's signal into a vector of anomalous and non-anomalous parts by using previous labeled data. The vector is then used with a linear transformation which models how much each part contributes to the signal, giving a power estimate. The power anomaly estimates are then combined with topology information to find routers that have many links with high anomaly power. BasisDetect was compared with exponentially weighted moving average and Fourier thresholding techniques in a network backbone router. BasisDetect was able to identify anomalous behavior with 50% less false positives. BasisDetect was also able to detect anomalies in a full sized network with 65% less false positives than a spatial anomaly detection technique.

[28] introduces a failure detection tool for IP networks named Shrink. It forms Shared Risk Link Groups (SRLGs) which are groups of links that all go down if one of the links fail. A Bayesian network is generated to check which of the SRLGs is most likely failing. They retrieve the status of the links which are either alive or dead. Then this information is used to infer which of the SRLGs is the problem. Shrink also takes into account marginal probabilities i.e some links/SRLGs are more likely to fail and there may be some misinformation or missing information in the SRLG description. The results are compared to two other alternatives BayesNet and MinSetCover. Shrink is able to detect the most likely cause of failure with a higher success rate than the alternatives, with a success rate of about 90% to 100% depending on the number of errors in the SRLG description.

[29] and [30] use received optical power and bit error rate measurements to uncover anomalous behavior. The anomalous behavior includes: signal overlap, tight filtering, gradual drift, cyclic drift and inter-channel interference. [29] used an algorithm based on Bayesian networks to achieve high prediction rates of 99.2% for a normal state, and 100% for tight filtering and inter-channel interference. Whereby [30] proposed two algorithms, the Bit-Error Rate Anomaly Detection (BANDO) and a probabilistic failure identification algorithm named LUCINDA. BANDO uses nodes within the network to monitor bit-error

rates and received optical power to determine degrading bit-error ratios. BANDO is able to send notifications containing useful information to LUCINDA. LUCINDA's main purpose is to act as a controller for the network and identify network failures using BANDO notifications. LUCINDA will return to the user the most probable cause of failure from a set of failure classes. In simulation this model was able to identify signal overlap and tight filtering with 100% accuracy, gradual drift with up to 70% accuracy and cyclic drift with up to 52% accuracy. It was also able to predict bit-error rate degradation several days before it happened.

[31] proposes an efficient algorithm to detect both "soft" and "hard" failures in a WDM optical network, using information that can be retrieved from the WDM layer such as interface error counters to locate failures. This algorithm locates "hard' failures using a binary tree, the leaves of this tree contain how channels relate to nodes in the network. Filling out this binary tree can be rather expensive so a pre-computation phase was incorporated to fill out the leaves of the binary tree before an alarm so that when the algorithm receives an alarm it simply traverses the binary tree to find likely candidates. For "soft" failures they are relying on bit error rate, signal to noise ratio and the number of discarded IP packets thresholds to determine when to send alarms to the algorithm. This is a very general algorithm since it does not specify the exact fault, which could lead to more troubleshooting.

[32] uses a model of many Intrusion Detection Agents (IDAs) which are composed of event processors, statistical processors, neural network classifiers and a post processor. These IDAs are chained together to detect intrusions and general anomalous behavior in a network. The statistical processor compares the PDFs of the recorded network data and the reference of "normal" behavior in the network. The PDF similarity is computed using the Kolmogorov-Smirnov (K-S) test. There are several different classes that are modeled as normal either over a specified time interval or between events. After being compared the statistical processor generates a vector to be used as inputs into the neural network for classification. With each new vector that is generated it updates the reference model to keep refining it so the neural network can make more accurate predictions in the future.

Then after the classification is completed the IDA will propagate its result to other IDAs. The results from simulations show that this method can reliably detect anomalous behavior when the anomalous behavior only makes up 3-5% of the data captured.

[33] suggested a network design pattern to reliably communicate large amounts of data reliably. Soft-failures, such as contaminated fiber connectors, can cause poor performance in the network and can go undetected for several months or longer. TCP will interpret lost packets due to soft-failures as congestion and reduce throughput to compensate. It is proposed to use Data Transfer Nodes (DTNs), generally Linux based servers, that are for the explicit purpose of transferring large amounts of data. There is some tuning required to make sure that there are no misconfigurations causing poor performance. It is also recommended to combine DTNs with performance monitoring software, such as PerfSONAR, to isolate problems that may be causing poor performance. Using those two techniques along with careful placement of network components can increase network efficiency. There are multiple use cases ranging from data centers to universities that show improvements following this design pattern.

[34] created a device to inspect optic cable end-faces, particularly in military avionics settings, using image classification. 2-D images are collected using CMOS sensor arrays, which are fed into a classification algorithm. The classification algorithm compares the image with two different standards the iNEMI (used by AT&T) and the Department of Defense (DoD) standard. Both standards split the end-face into zones that have a range of imperfections that are acceptable to minimize the risk of failure. This allows for field operators with no expertise to accurately asses the cleanliness of the end-face and replace/clean them if needed. A depth map is also generated to detect if the imperfections are degradation or contamination, so the user can take the appropriate action.

2.8 Data Analysis

With large amounts of data being produced, it is important to be able to analyze this data and derive meaning from it. There are many ways that this can be accomplished, by looking at patterns/trends, checking the statistical data such as variance and machine learning. A first good step is to do Exploratory Data Analysis (EDA). EDA is the process of understanding the dataset either through visual or statistical exploration. There are many tools to help accomplish this goal. The correlation matrix can be computed in order to see the relationship between each pair of features, which in turn can help you do feature reduction. The distribution of the values can be visualized to see which machine learning technique is best suited to this dataset or if the dataset needs to be normalized/scaled.

2.8.1 Machine Learning

Machine learning leverages the computational power of modern hardware to classify new data entries or predict a value of a future entry in a set of data. The benefit of it is that you do not have to explicitly program how to achieve the desired result, rather you program how the machine should learn how to predict or classify. Machine learning can be separated into two classes, supervised and unsupervised learning. The difference between supervised and unsupervised learning is whether or not you need training data. Training data is sample data similar to the data the algorithm will be working with except it has the correct labels attached to them. With this information you can train a classification or regression model.

2.8.2 Supervised Learning

As mentioned before supervised learning is when the machine learning technique requires training data to train a model to perform regression or classification. Classification is when the algorithm is given labels that do not have an inherent ordered value i.e categories and predicts a new entry based on the training data examples, such as ”apple”, ”orange” and

"banana" if you wanted to classify fruits. On the other hand, regression has features that do have an ordered value such as length. A couple of supervised machine learning techniques are K-nearest neighbors (KNN), decision trees, support vector machines (SVM), logarithmic regression and the naive Bayes classifier. The two used in this work are KNN and decision tree classifiers to try and predict the state of the connector which will be labeled either "contaminated" or "clean". The K-Nearest Neighbors algorithm classifies a new data entry based on the majority classification of the nearest K entries in training data. An example is shown in Figure 2.18, the K value for this example is set to 3 and there are only 2 features that are considered. The training data has 6 entries which are all either labeled as "clean" or "contaminated". The new entry's 2 out of the 3 closest neighbors' label are "clean" therefore the new entry will be labeled as "clean". KNN classifiers can be used with more features and use different distance measures i.e non-euclidean to classify new entries. Decision trees use the training data to create a tree of branching paths which will be traversed by a new data entry for prediction. A trivial example of a decision tree for classification is shown in Figure 2.19.

Over-fitting can present a serious problem when training machine learning models. Over-fitting is when a model is too specifically trained to the training data set so the model will not generalize well. If a model is over-fitted to the training data it will perform well when used on the training set however, when new data is presented to the model it will have poor performance. This is due to the model being to specifically tuned to the noise/anomalies of the training set not allowing it to generalize well to data it's never seen before that may not have similar anomalies or noise.

2.8.3 Unsupervised Learning

Unsupervised learning techniques do not require labeled training data to train a model and are used to derive an association between two items or find clusters of data. Association mining is used to find patterns in data sets, the classical example is to find products that are bought together most frequently. Algorithms that accomplish this are the apriori algorithm

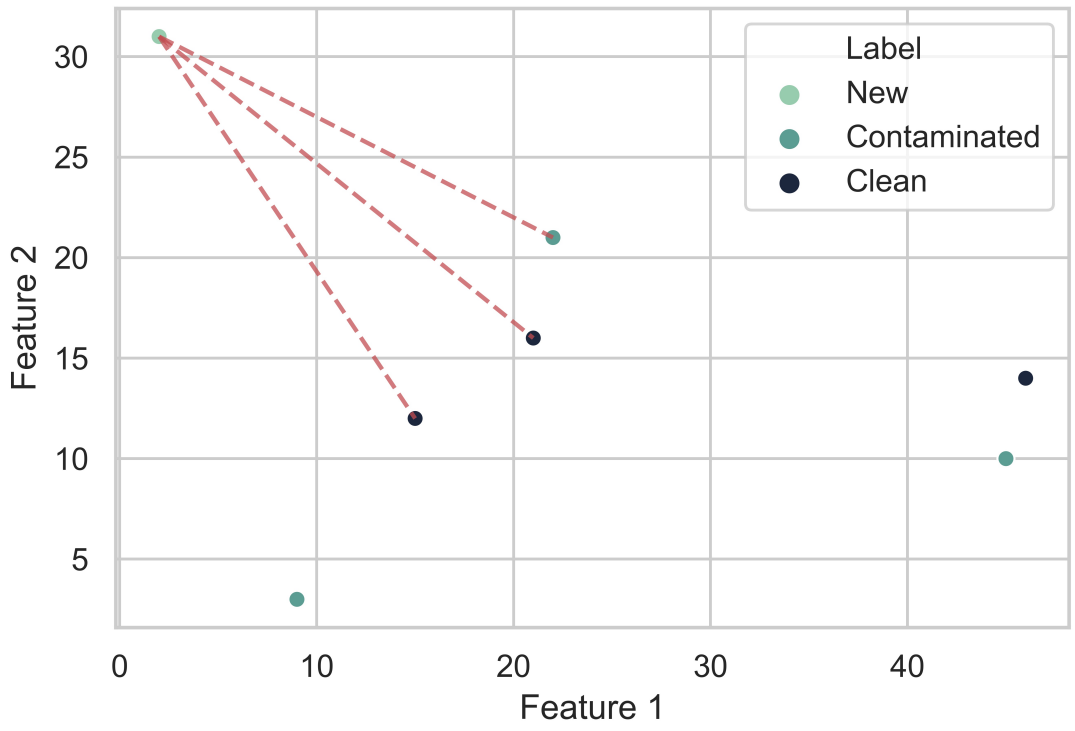


Figure 2.18: K-Nearest Neighbors (KNN) example

and the frequent pattern growth. Clustering finds inherent groups of data points based on the similarity of their features. K-means, Density-based spatial clustering of applications with noise (DBSCAN) and agglomerative hierarchical clustering achieve this. K-means clustering takes a user-defined number of clusters to form, K. K centroids are placed in the feature space randomly, each data point is assigned to the cluster corresponding to the centroid that is the closest. The centroids are then moved to the center of the points that belong to that cluster. Then the points are reassigned to the cluster centroid that is closest. This process continues until there are no changes to the cluster assignments of the points.

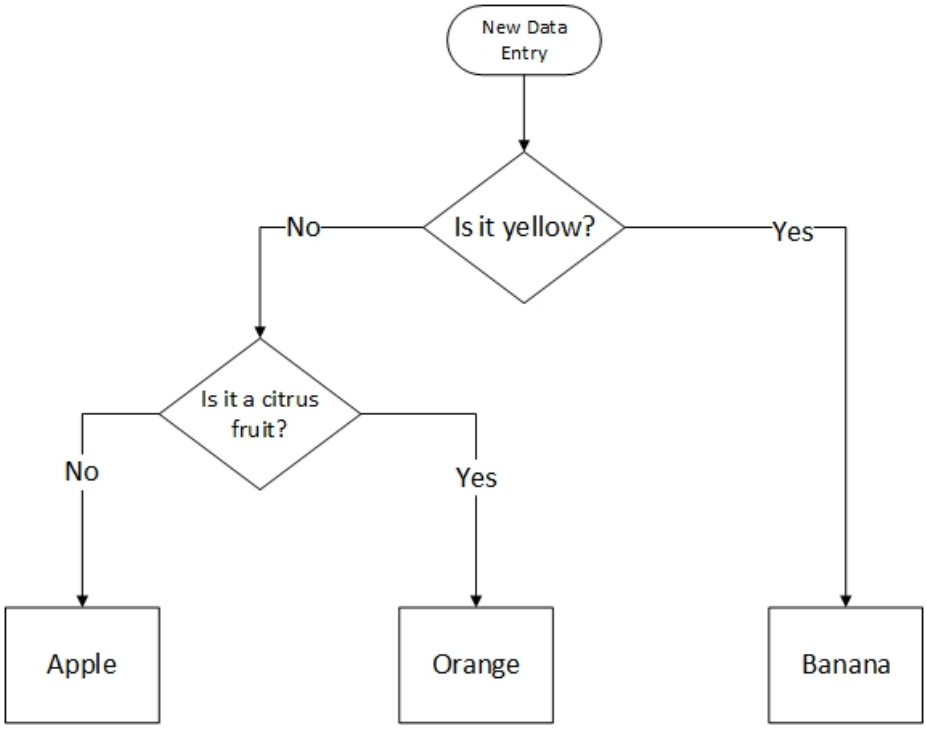


Figure 2.19: Decision tree example

2.8.4 Preprocessing

In many cases to get the most out of machine learning algorithms data preprocessing must be implemented. The most common preprocessing steps that need to be performed are cleaning and scaling. Cleaning the data requires that unnecessary features and entries with missing data are removed. An alternative to removing the entries with missing data is to fill the missing data in with a value. For instance, if one of the samples is missing one feature's data, you could remove it from the data set or replace the null value with the mean of the same feature of other samples. This means that you can keep the data and not skew the results by having empty entries. Scaling is also important because many machine learning techniques, such as k-nearest neighbors classifiers, take the magnitude of the input data into account. Thus if the scale of one of the features is much larger than the other it will have a much more significant impact on the result than the other features. Also similar to scaling normalizing changes the distribution to be normal, i.e 0 mean and

Table 2.6: One hot encoding

Entry	Fruit	Apple	Orange	Banana
0	Apple	1	0	0
1	Orange	0	1	0
2	Banana	0	0	1

variance of 1, which is also done to increase the effectiveness of certain machine learning techniques. Data engineering is also a helpful tool to optimize the performance of models. Data engineering is when you use existing features to create a new more useful feature that the model can use. Example, if the dataset contains the start time, stop time and bits sent you could calculate throughput by:

$$Throughput = \frac{Bits\ sent}{(Start\ time - Stop\ time)} \tag{2.8}$$

And dependent on the application having the throughput could be a much more useful feature than the other three separately. Depending on the type of application, it may be necessary to encode categorical values. A common scenario when you would need to do this is when using neural networks. Since neural networks cannot take categorical data as inputs you must encode it into a number value. A common encoding scheme is one-hot encoding which is done by taking the categorical sample space and create a new feature for each unique value or omitting one entry. Going back to the fruit labels, if we have "apple", "orange" and "banana" as the sample space for labels you could one hot encode every entry as shown in Table 2.6. You can also omit one entry and if all other values are 0 then it is the non-included label as shown in Figure 2.7.

2.8.5 Hyper-parameter tuning

Hyper-parameters are values that affect how the machine learns and are not directly related to the input data. Some examples are the number of nodes in a layer of a deep neural

Table 2.7: One hot encoding without last column

	Fruit	Apple	Orange
0	Apple	1	0
1	Orange	0	1
2	Banana	0	0

network or the K value for the KNN classifier. Hyper-parameter tuning is the process of finding the combination of values for each hyper-parameter which yields the best performance. Two common techniques to tune hyper-parameters are grid searches and random searches. A grid search will compare the performance of the model trying different specified hyper-parameter values. Random search picks a random point in the set of possible points to be the best value, then randomly chooses a new value near the best value to compare performance with. If the new value does better it becomes the best value. This pattern continues until a stop condition is met.

Chapter 3

Methodology

The focus of this work is to detect contaminated optical connectors through analysis network statistics that can be collected with no added hardware. As shown in the related work, optical power is reduced by contamination on the connectors. This presents an opportunity to leverage the optical power measurements provided by SFPs to detect contamination remotely with no additional hardware. As a preface for detecting contaminated fibers using SFPs it is important to consider the accuracy of the SFP's DOM capabilities. The SFF-8472 outlines the acceptable accuracy for each statistic provided by DOM. Figure 3.1 shows the standard tolerance range of the SFP optical power measurements and Figure 3.2 shows the possible loss induced by the inaccuracy of the SFP optical power measurements. This tolerance range is much too wide to be useful in practice. In actuality this tolerance range is device specific and is very likely to be much smaller than the proposed ± 3dB, however on the data sheet they will not advertise this and simply put that it adheres to the SFF-8472 standard.

The main objectives of this work are to:

- Measure the accuracy of SFP optical power data.
- Find a framework to detect contaminated connectors, using SFP optical power data.
- Detect contamination using Layer 2, 3 and 4 network statistics.

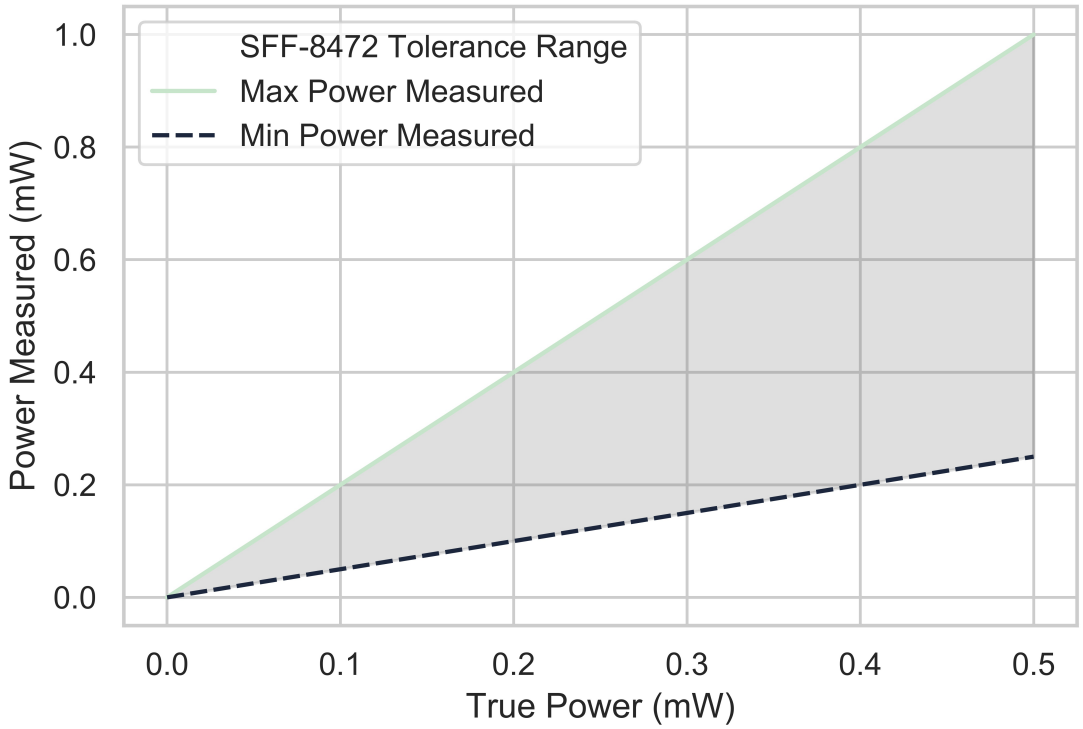


Figure 3.1: SFP tolerance range standard. The shaded area represents the acceptable range of accuracy.

3.1 Accuracy of SFP Optical Power Data

In order to find the true accuracy range of the SFPs, a series of experiments were conducted where the values of a dedicated power meter were compared to that reported by the SFP DOM through the switches. The experimental setup is shown in Figure 3.3 and shows the tolerance of the equipment used.

3.1.1 Test Cases

The following measurements were made to test the accuracy of two SFPs. Each of the following values was compared to the value read by the power meter:

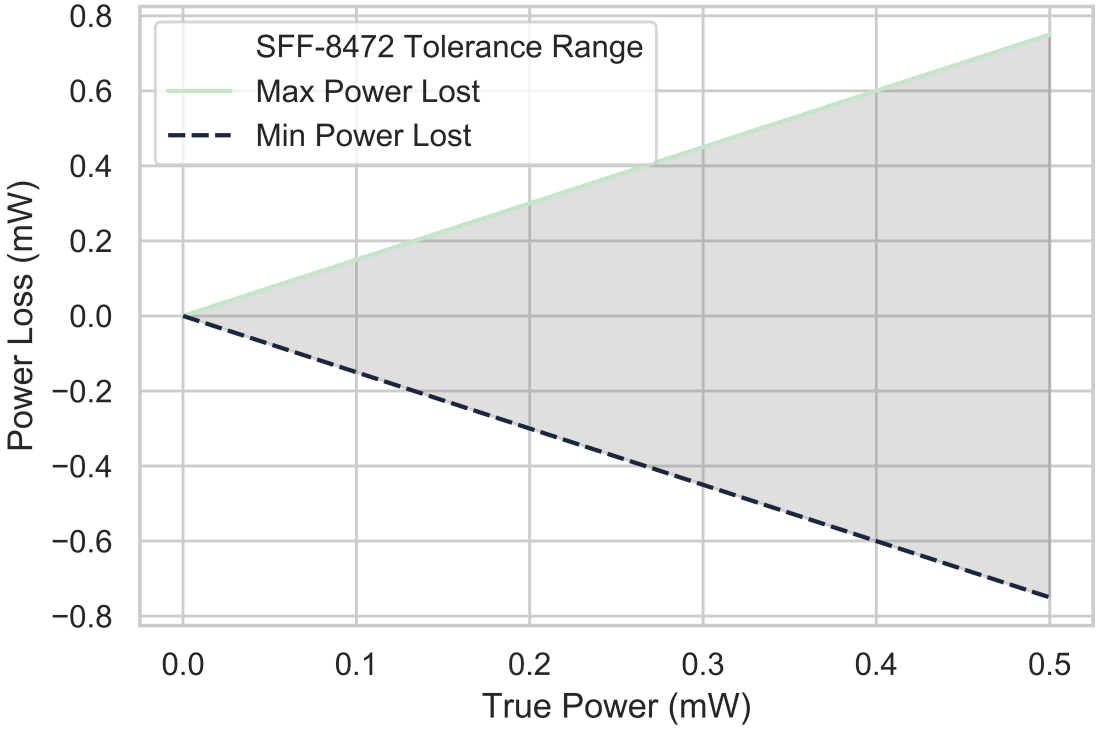


Figure 3.2: Loss induced by tolerance range standard. The shaded area represents the possible loss range.

3.1.2 Tolerance adjustment

It is important to account for all possible power losses in this setup to find the true accuracy of the SFP. This means that we have to take into account every factor that could possibly affect the optical power. Thus the loses when using the experimental setup in Figure 3.3 that need to be accounted for are:

- $T_C = 1$ - Power loss due to the cable during Tx readings ($\leq$ 0.3 dB = 7%)
- $T_A = 1$ - Power loss due to the adapter for the power meter ($\leq$ 0.3 dB = 7%)
- $T_M = 1 \pm$ Power meter inaccuracy ($\pm$ 0.211 dB = 5%)

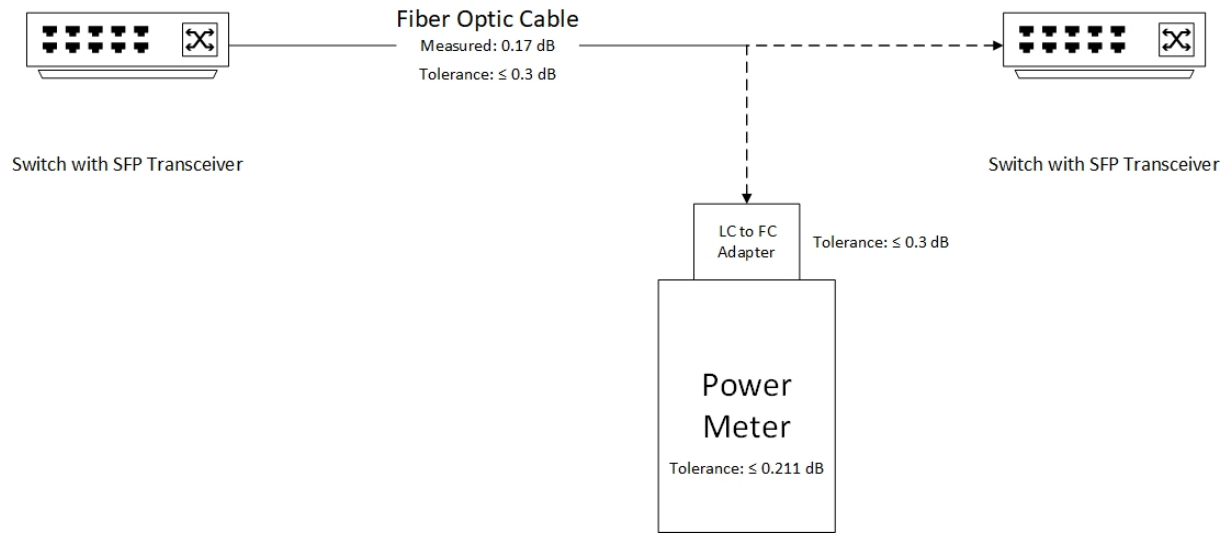


Figure 3.3: Power meter connected to SFP via fiber cable to measure accuracy.

All tolerances listed are from the manufacturer's specifications. If the $Power\ Meter\ Measurement = P_M$ and $SFP\ Measurement = P_S$ then the max difference for Tx readings can be described as:

$$max(\delta_{Tx}) = max(P_S - P_M T_C T_A T_{M+}, P_S - P_M T_C T_A T_{M-}) \tag{3.1}$$

and the max difference for Rx readings can be described as:

$$max(\delta_{Rx}) = max(P_S - P_M T_A T_{M+}, P_S - P_M T_A T_{M-}) \tag{3.2}$$

However if you know the measured insertion loss of the cable or the adapter it is better to use that number.

The power loss due to the cable was not considered during the Rx readings do to both measurements being taken at the end of the cable. Whereas for the Tx readings the power meter reading was at the end of the cable and the SFP reading did not have the insertion loss due to the cable so we must adjust the values for this.

Table 3.1: SFP Accuracy Test Cases

Tx power of clean SMF connectors using a Fiberstore SFP
Rx power of clean SMF connectors using a Fiberstore SFP
Rx power of contaminated SMF connectors using a Fiberstore SFP
Tx power of clean SMF connectors using a Finisar SFP
Rx power of clean SMF connectors using a Finisar SFP
Rx power of contaminated SMF connectors using a Finisar SFP
Tx power of clean MMF connectors using a Fiberstore SFP
Rx power of clean MMF connectors using a Fiberstore SFP
Rx power of contaminated MMF connectors using a Fiberstore SFP
Tx power of clean MMF connectors using a Finisar SFP
Rx power of clean MMF connectors using a Finisar SFP
Rx power of contaminated MMF connectors using a Finisar SFP

3.2 Detecting Contaminated Connectors

To study the effects of contaminated fiber optic connectors a series of experiments were conducted using the topology shown in Figure 3.4. There are two hosts both running iPerf3 on Ubuntu 18.04 transferring data across one fiber optic link while a remote host monitors and records the Tx and Rx values provided by the two switches/SFPs.

3.2.1 Contamination and Cleaning Processes

To contaminate the end faces of the optical connectors in the case of oil, the connectors were gently touched with a fingertip multiple times to ensure that there was an oil residue as shown in Figure 3.5 (a). To contaminated the connectors in the case of dust, fine grain sand was applied to both sides of the fiber optic cable using a swab that resulted in particles blocking the core as shown in Figure 3.5 (b).

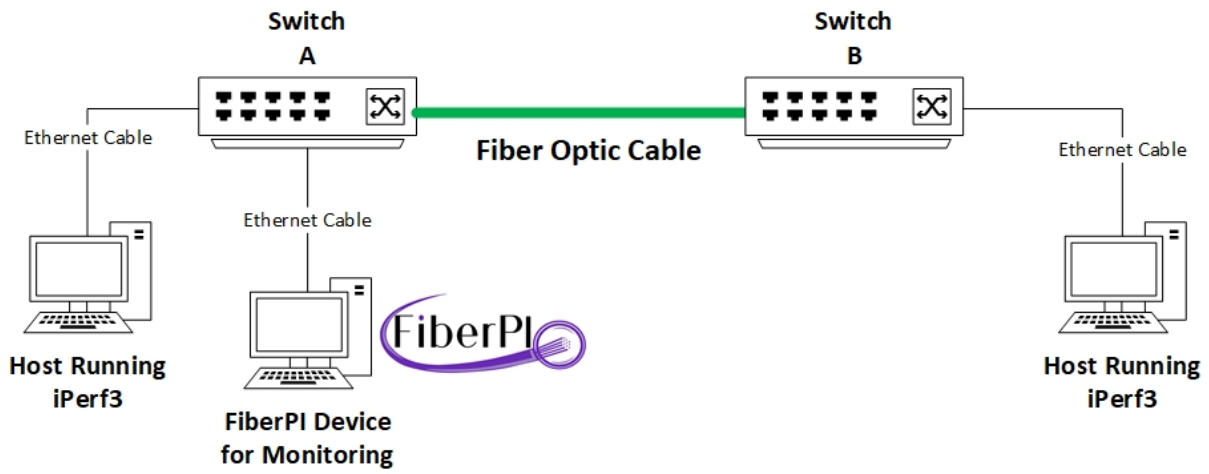
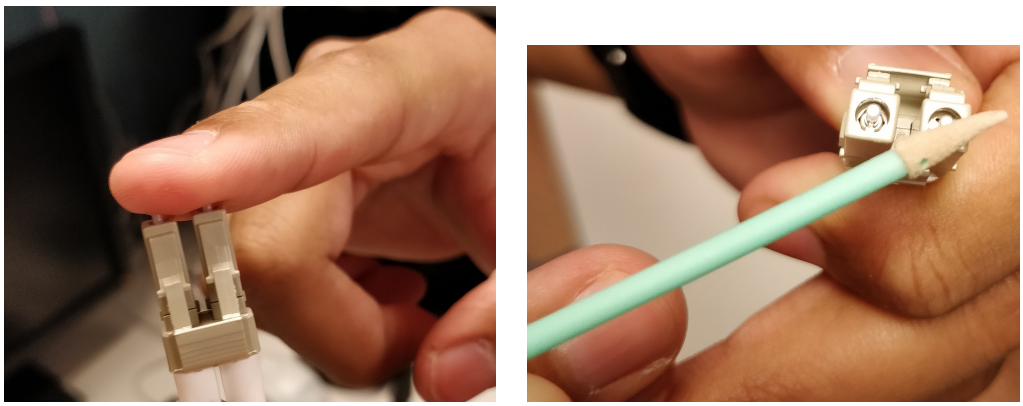


Figure 3.4: Topology of experimental network



a) Fingerprint contamination process

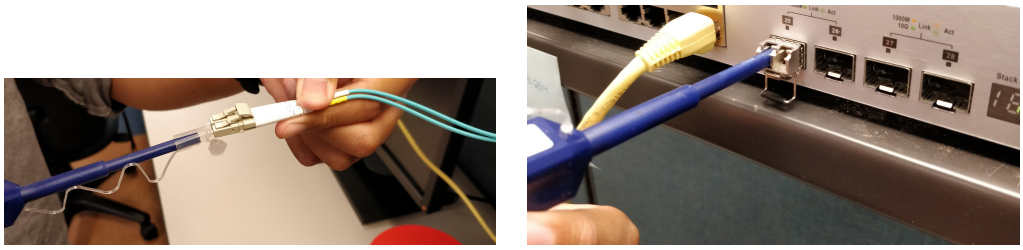
b) Dust contamination process

Figure 3.5: Contamination process

To clean the cable a one-push pen was used as shown in Figure 3.6 (a). To clean the SFP connectors the same one-push pen was used as shown in Figure 3.6 (b).

3.2.2 Monitoring

The monitoring was conducted using FiberPI a software developed specifically for this work. It is implemented using Python which will poll switches for DOM statistics, which include Tx and Rx power levels when directed to.



a) Cable cleaning process

b) SFP cleaning process

Figure 3.6: Cleaning process

3.2.3 Test Cases

For each test case the Tx and Rx values of both SFPs will be collected every 30 seconds over a 24 hour period, this will provide 2880 data points for each set. For each tuple of Tx and Rx value pairs the difference can be calculated to give the power lost between transmission in both directions. Samples of the power difference provided through FiberPI were analyzed for 6 separate test cases:

- Clean SMF Link
- SMF Link with Fingerprints
- SMF Link Contaminated with Dust
- Clean MMF Link
- MMF Link with Fingerprints
- MMF Link Contaminated with Dust

3.3 Top-Down Method

It is preferable to monitor the network for symptoms higher up the OSI data stack because it is too taxing to be consistently polling every single SFP in the network to make sure that they are clean. Thus we turn to layer 2, 3 and 4 statistics. Layer 2 will be monitored

using ifconfig, layer 3 will be monitored by nstat and layer 4 will be monitored using iPerf and nstat. A list of the statistics that were considered are listed in Appendix A. The motivation for using a layered approach is to leverage machine learning techniques in order to detect poor network performance that could be a symptom of contaminated links in the network. Once poor performance is detected using machine learning then Inequality 4.14 can be used to confirm our suspicions. This should alleviate the problem of polling overhead and allow this detection system to scale better.

Similar to the other experiments the topology in Figure 3.4 will be used. The two test cases will be when the fiber link is contaminated with dust and when the link is clean. The Layer 2, 3 and 4 statistics listed in Appendix A will be collected. Filtering will be done to the samples due to the feature space being much too large. Principle Component Analysis (PCA) and choosing the features with the highest normalized variances can be used to find the features that are affected the most by the contaminated link. Then a handful of machine learning techniques will be trained using the Python library Scikit-Learn to attempt to classify new observations.

Chapter 4

Results and Discussion

4.1 SFP Accuracy

The results of comparing the values given by the power meter to the values given by the SFPs DOM capability are summarized in Table 4.1. The optical power meter measurement range is in this format, *measured power; (minimum power, maximum power)*. The range in parentheses accounts for the power meter accuracy tolerance of ± 0.211 dB. Similarly, the adjusted value range shows the power meter measurement adjusted in this format, *adjusted power; (minimum power, maximum power)*. Where the range in parentheses accounts for the adapter and cable loss for Tx measurements only. Then the max difference can be computed by using the largest difference between the SFP measurement and one of the adjusted value ranges which is in bold.

Table 4.1: SFP Transceiver Accuracy

Test	**Optical Transceiver Measurement (mW)**	**Optical Power Meter Measurement Range (mW)**	**Adjusted Value Range (mW)**	**Max Difference (mW)**	**Max Difference (dB)**
{Tx, SMF, Fiberstore, Clean}	0.247	0.226; (0.215, 0.237)	0.252; (0.239, **0.264**)	0.017	0.277
{Rx, SMF, Fiberstore, Clean}	0.222	0.24; (0.228, 0.252)	0.257; (0.244, **0.270**)	0.048	0.711
{Rx, SMF, Fiberstore, Contaminated}	0.148	0.16; (0.152, 0.168)	0.171; (0.163, **0.180**)	0.032	0.711
{Tx, SMF, Finisar, Clean}	0.284	0.314; (0.298, 0.330)	0.350; (0.332, **0.367**)	0.083	0.888
{Rx, SMF, Finisar, Clean}	0.224	0.242; (0.230, 0.254)	0.259; (0.246, **0.272**)	0.048	0.709
{Rx, SMF, Finisar, Contaminated}	0.148	0.160; (0.152, 0.168)	0.171; (0.163, **0.180**)	0.032	0.711
{Tx, MMF, Fiberstore, Clean}	0.273	0.243; (0.231 ,0.255)	0.271; (**0.257**, 0.284)	0.016	0.258
{Rx, MMF, Fiberstore, Clean}	0.247	0.255; (0.242, 0.268)	0.273; (0.260, **0.287**)	0.040	0.565
{Rx, MMF, Fiberstore, Contaminated}	0.160	0.167; (0.159, 0.175)	0.179; (0.170, **0.188**)	0.028	0.601
{Tx, MMF, Finisar, Clean}	0.208	0.227; (0.216, 0.238)	0.253; (0.240, **0.266**)	0.058	0.852
{Rx, MMF, Finisar, Clean}	0.210	0.239; (0.227, 0.251)	0.256; (0.243, **0.269**)	0.059	0.860
{Rx, MMF, Finisar, Contaminated}	0.156	0.168; (0.160, 0.176)	0.180; (0.171, **0.189**)	0.033	0.699

The results are clear, the ± 3 dB standard is not realistic at all. For all cases, where we took the worst case scenario, all had a max difference of < 1 dB. Considering this was the case for both brands of SFP using 4 different inexpensive SFPs on both types of fiber, it seems that they are much more accurate than the standard would have you believe. This is important as the following framework will take into account these possible inaccuracies.

4.2 Contamination Detection

Now that the inaccuracies of the SFP DOM optical power measurements have been explored, we can use this information to create a framework that can identify contamination that affects optical power, i.e contamination that is causing insertion loss. This will be achieved by finding the threshold where it is certain that there is contamination causing insertion loss by taking into account all other factors.

4.2.1 Contamination Detection Method

It is important to note that contamination is not the only factor that will induce insertion loss, so that must be taken into account as well in the framework. To find this threshold we must get the expected power loss due to non-contaminant factors (α) and bias it by the maximum possible power loss due to the SFP transceiver inaccuracies. The maximum power loss due to inaccuracies occurs when the actual or "true" power difference (δ) is greater than the measured power difference ($\hat{\delta}$). There are 3 inputs that will be needed to find an accurate threshold which are:

- $\lambda_{Tx} =$ | Tolerance range of Tx SFP Transceiver (dB) |
- $\lambda_{Rx} =$ | Tolerance range of Rx SFP Transceiver (dB) |
- $\Delta =$ Expected attenuation due to non-contaminant factors (dB)

Δ is calculated by adding all the loss you expect to see in the tested link due to non-contaminant factors such as attenuation due to distance, optical splitters, connection adapters, etc. Using Δ you can compute α with this equation:

$$\alpha = \hat{t}\left(1 - 10^{\frac{-\Delta}{10}}\right) \tag{4.1}$$

Where $\hat{t}$ is the measured Tx power.

Example: Assume you have a 50/50 optical (3 dB loss) splitter in the middle of a 10 km single mode optical fiber which has attenuation per km of 0.5 dB/km and an insertion loss of 0.3 dB. Add up all the loses in dB to get delta:

$$\Delta = 0.3 + 0.5 \cdot 10 + 3 = 8.5\ dB \tag{4.2}$$

Then use this Δ to compute α,

$$\alpha = \hat{t}\left(1 - 10^{\frac{-8.5}{10}}\right) = 0.1479\hat{t} \tag{4.3}$$

The rule to detect contamination can be given by:

$$\hat{\delta} > \alpha + max(\delta - \hat{\delta}) \tag{4.4}$$

As a note, we do not need to consider $(|\delta - \hat{\delta}|)$ due to the fact that $(\delta - \hat{\delta}) \geq (\hat{\delta} - \delta)$. To show this, because $\hat{\delta}$ is a constant:

$$max(\delta - \hat{\delta}) = max(\delta) - min(\hat{\delta}) = max(\delta) - \hat{\delta} \tag{4.5}$$

and

$$max(\hat{\delta} - \delta) = max(\hat{\delta}) - min(\delta) = \hat{\delta} - min(\delta) \tag{4.6}$$

The true power loss can never be less than 0, since there are no amplifiers, thus $min(\delta)$ is capped at 0 then and lowest value for $max(\delta)$ is 0 thus,

$$max(\delta) \geq -(min(\delta)) \tag{4.7}$$

which means,

$$max(\delta - \hat{\delta}) \geq max(\hat{\delta} - \delta) \tag{4.8}$$

moving forward with the right hand side of inequality 4.4 we can expand it to:

$$\alpha + max(\delta - \hat{\delta}) = \alpha + max((t - r) - (\hat{t} - \hat{r})) \tag{4.9}$$

where t = true power transmitted, r = true power received, $\hat{t}$ = measured power transmitted and $\hat{r}$ = measured power received. Because $\hat{t}$ and $\hat{r}$ are constants equation 4.9 becomes:

$$\alpha + max(t - r) - (\hat{t} - \hat{r}) = \alpha + max(t) - min(r) - \hat{t} + \hat{r} \tag{4.10}$$

Where max(t) is the largest value for Tx power that the SFP optical power data will show and min(r) is the smallest value for Rx power that the SFP optical power data will show:

$$max(t) = \hat{t}\left(10^{\frac{\lambda_{Tx}}{10}}\right) \tag{4.11}$$

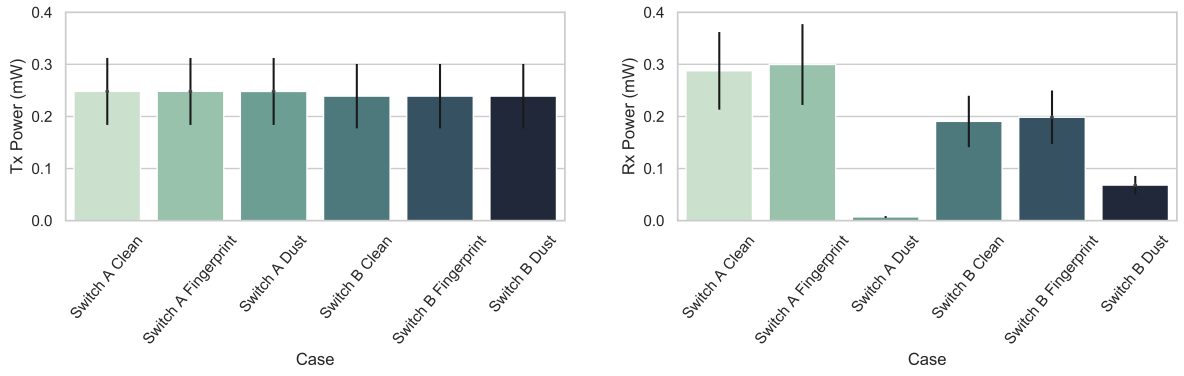
$$min(r) = \hat{r}\left(10^{\frac{-\lambda_{Rx}}{10}}\right) \tag{4.12}$$

Substituting equations 4.1, 4.11 and 4.12 and into 4.10:

$$\hat{t}\left(1 - 10^{\frac{-\Delta}{10}}\right) + \hat{t}\left(10^{\frac{\lambda_{Tx}}{10}}\right) - \hat{r}\left(10^{\frac{-\lambda_{Rx}}{10}}\right) - \hat{t} + \hat{r} \tag{4.13}$$

Finally, using 4.13 as the right hand side of inequality 4.4 and simplifying, a connector is contaminated if:

$$\hat{\delta} > \hat{t}\left(10^{\frac{\lambda_{Tx}}{10}} - 10^{\frac{-\Delta}{10}}\right) + \hat{r}\left(1 - 10^{\frac{-\lambda_{Rx}}{10}}\right) \tag{4.14}$$



a) SMF Tx Power

b) SMF Rx Power

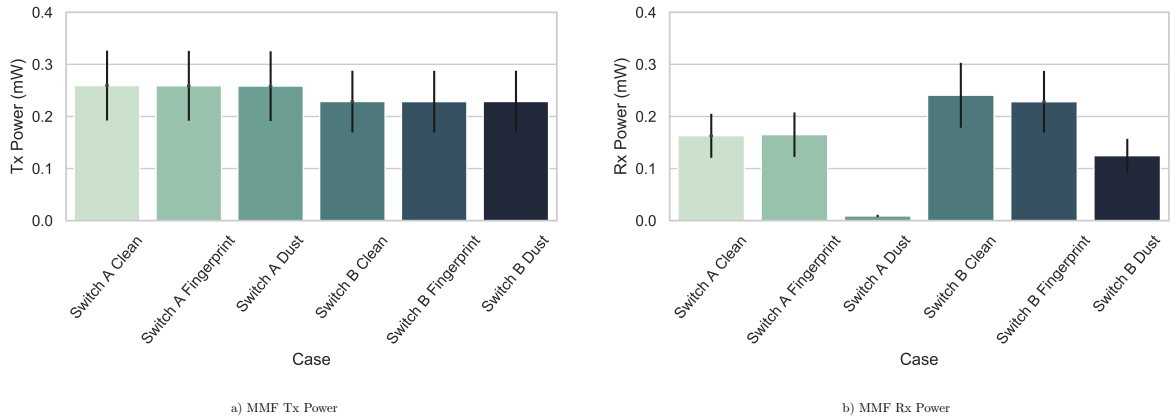
Figure 4.1: SMF Power Measurements with ± 1 dB error bars

4.2.2 Contamination Detection Results

After a day of running, 2880 samples for each test case were collected and analyzed. Figure 4.1 shows the Tx power and Rx power for both switches using single-mode connectors. Each bar shows the mean of the power measurements with the error bar showing the more realistic ± 1 dB. Thus the true power must be within the range of the error bar. The confidence intervals were computed but were too small to include in the figure. It acts as expected even over a long period of time, 24 hours. The Tx values all stay roughly the same, independent of the state of contamination. The error bars are also near identical due to the power levels being similar. When the connectors only have fingerprints, the results are inconclusive, there is no significant optical power change caused by fingerprints, a trend that will continue. However, the Rx power on both switches have a significant decrease in optical power due to the dust. The dust also has a smaller error bar due to having a small value being read from the optical transceiver.

Figure 4.2 shows the power measurements collected using multi-mode connectors. These measurements follow the pattern presented by the single-mode measurements. The Tx power stays the same independent of the contamination status, while the Rx values only see a significant change when contaminated with dust.

Now we can use those 2880 samples to compute the insertion loss i.e power loss. Using



a) MMF Tx Power

b) MMF Rx Power

Figure 4.2: MMF Power Measurements with ± 1 dB error bars

these values we can evaluate the measurements using Inequality 4.14, where $\Delta = 0.3dB$ and $\lambda_{Tx} = \lambda_{Rx} = 1dB$. Figure 4.3 shows the mean power loss from switch A to switch B (Switch A Tx - Switch B Rx) and from switch B to switch A (Switch B Tx - Switch A Rx). There are error bars in the figure, however there is almost no variation in the power lost over time causing them to be minuscule. There are red dashed lines for each case, where the inequality finds the connectors to be contaminated. The clean and fingerprint contaminated connectors do not cross this threshold meaning that we can not declare them contaminated because the power loss may be caused due to inaccuracies. The dust contaminated connector does cross this threshold by a wide margin, meaning that one of the two relevant connectors is contaminated. The power loss caused by the dust was 3 dB from A to B and almost all of the power from B to A, much higher than the anticipated 0.3 dB.

Figure 4.4 shows the mean power difference from switch A to switch B using single-mode connectors. Again, they have error bars which are the black dots in the figure, showing little to no variation throughout the testing period. A similar story as the past, the clean and fingerprint contaminated connectors did not cross the threshold, meaning that they are likely uncontaminated. The dust contaminated connectors pass the threshold by a wide margin again meaning that they are contaminated.

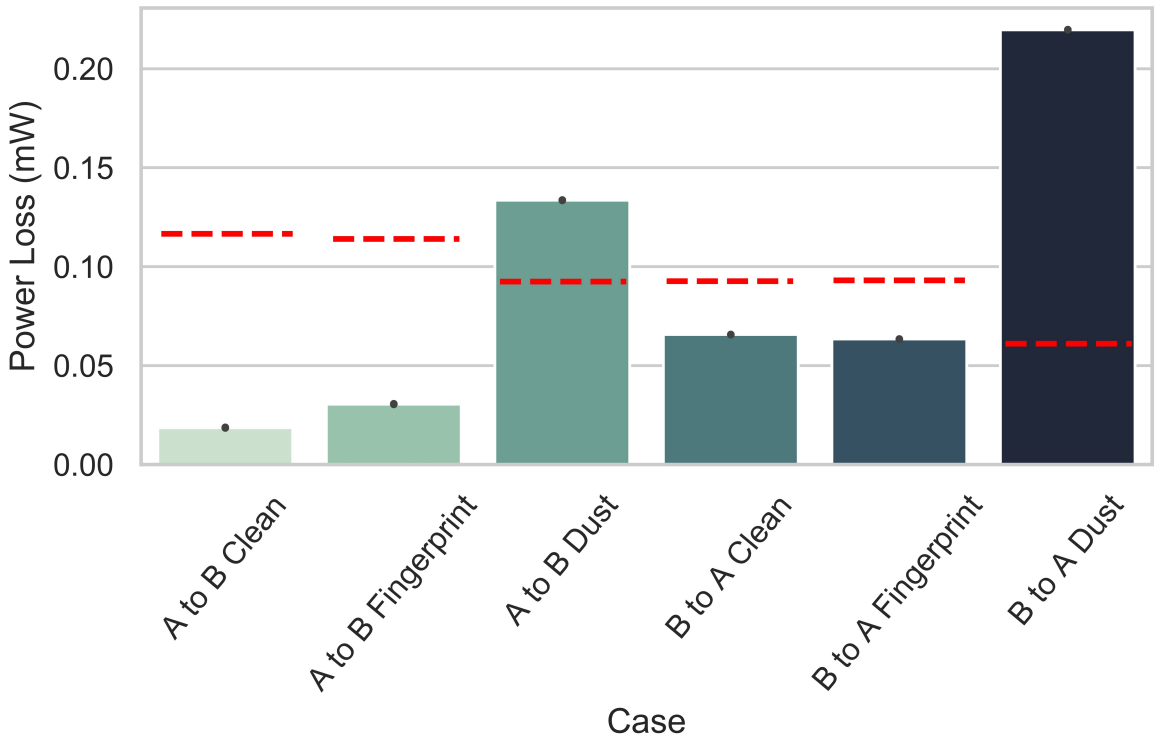


Figure 4.3: Power loss using single-mode fiber, where the dashed lines represent the contamination threshold for each case

There is something interesting when looking at the clean and fingerprint contaminated samples in Figure 4.4, the power loss was actually negative i.e a power gain. Using Figure 3.4 as the topology for this experiment it seems counter-intuitive that there is power gain, as there are no components in this network that amplify optical signals. The explanation for this is simply that the inaccuracies of the SFP are at play, it is possible that measured Tx power is at the lower bound of the $\pm$ 1 dB range and the measured Rx power is at the upper bound of the $\pm$ 1 dB range. Considering $P_{Tx} \approx P_{Rx} = P$ and $\lambda_{Tx} = \lambda_{Rx} = 1$:

$$\left(P \cdot 10^{-\frac{\lambda_{Tx}}{10}}\right) - \left(P \cdot 10^{\frac{\lambda_{Rx}}{10}}\right) = P(0.794 - 1.258) = -0.464 \cdot P \tag{4.15}$$

Thus, Equation 4.15 gives a negative value, showing that it is possible to get negative

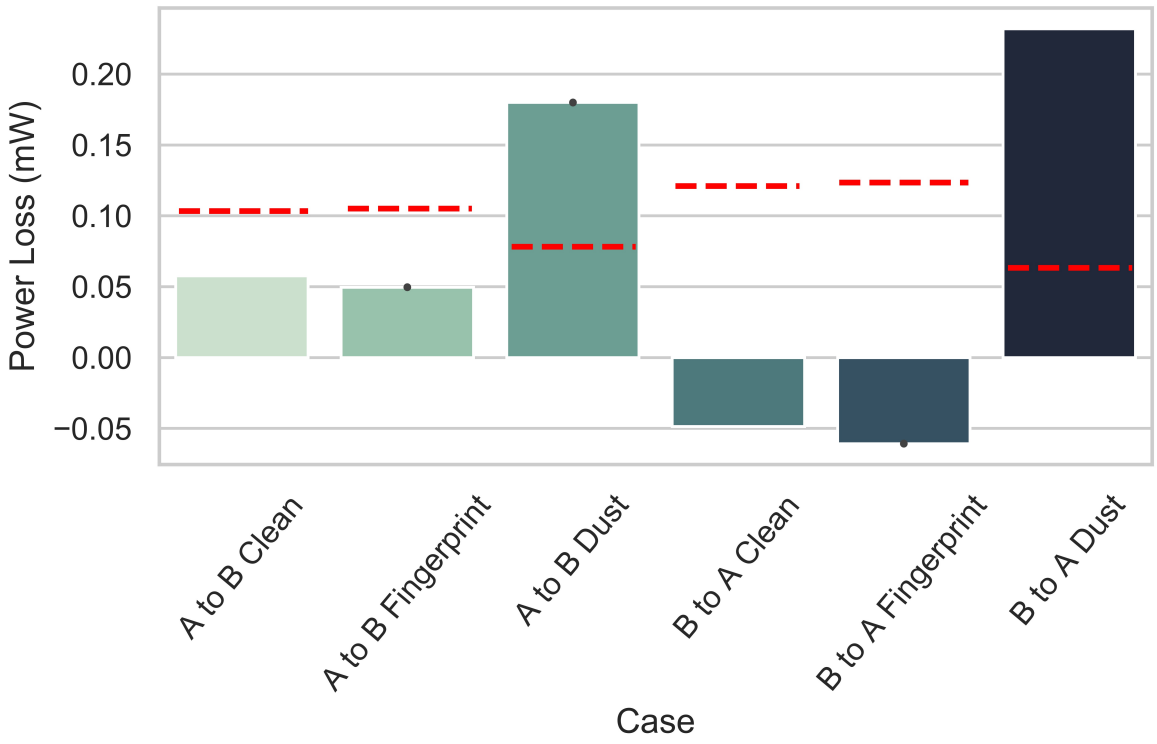


Figure 4.4: Power loss using single-mode fiber, where the dashed lines represent the contamination threshold for each case

values. In the end it works out, the mean of the clean and fingerprint contaminated samples do not cross the threshold meaning that they are not contaminated, which is why we can assume $P_{Tx} \approx P_{Rx}$. And finally, the dust contaminated connectors, cross the threshold detecting the final case as contaminated using the contamination rule.

As demonstrated by the examples Inequality 4.14 was able to detect all the contaminated connectors with no false positives, showing that it is an effective method to use DOM statistics provided by SFPs to detect contaminated connectors.

4.2.3 Contamination Score

To provide a more detailed description of the severity of contamination, rather than just if it is contaminated or not, it is proposed to use a score that increases as the contamination makes more of a difference on optical performance. We will normalize the value of the contamination score (CS) by setting 100% power loss as a contamination score of 100 by using a contamination coefficient (CC). CC and CS are given by:

$$CC = \frac{100}{10^{5(\hat{t}-\alpha)} - 1} \tag{4.16}$$

$$CS = max(0,\ CC \cdot (10^{5(\hat{\delta}-\alpha)} - 1)) \tag{4.17}$$

If T is the threshold for contamination given by:

$$T = \hat{t}\left(10^{\frac{\lambda_{Tx}}{10}} - 10^{\frac{-\Delta}{10}}\right) + \hat{r}\left(1 - 10^{\frac{-\lambda_{Rx}}{10}}\right) \tag{4.18}$$

Figure 4.5 shows how the number line is separated, the CS will provide a more intuitive description of power loss $> \alpha$.

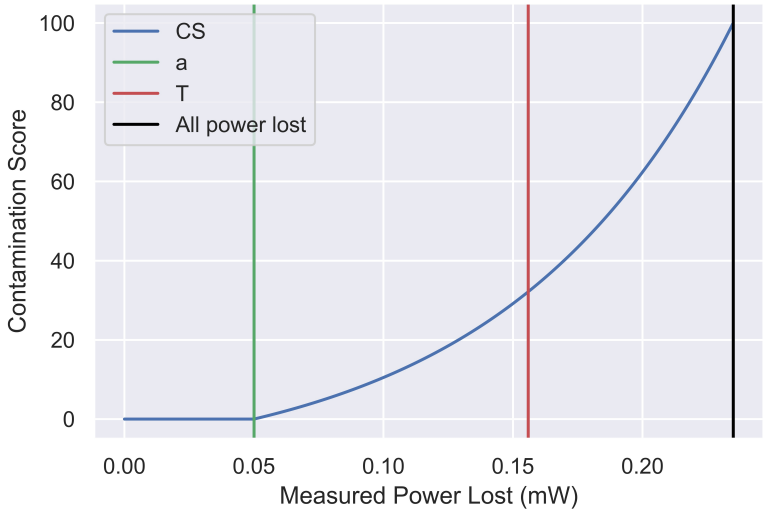


Figure 4.5: Power loss threshold visualization

A translation can be made from T to a CS score (τ_{cs}) which is given by:

$$\tau_{cs} = CC \cdot (10^{5(T-\alpha)} - 1) \tag{4.19}$$

If a CS $\geq \tau_{cs}$, then the connector is contaminated, given $\Delta, \lambda_{Rx}, and\ \lambda_{Tx}$ are accurate.

4.3 Top-Down Method

4000 samples were collected and analyzed with all the statistics shown in Appendix A. Then the features that did not change were removed, which was almost all of them. The only two significant features that changed were the retransmissions reported by nstat and the delayed acknowledgments reported by nstat. Using these features a decision tree classifier and a k-nearest neighbors classifier were trained.

4.3.1 Initial Analysis

Figure 4.6 shows a scatter plot of the data points collected with the contamination labels that are assigned to it based on the contamination score, where blue is clean and red is contaminated. Just through visual analysis, there seems to be a concentration of contaminated samples where the delayed acknowledgments and retransmission counts are higher. The correlation between retransmissions and contaminated links is intuitive due to the fact that some of the packets will be discarded due to errors in the TCP segments, meaning they will have to be retransmitted.

Figure 4.7 (a) shows the scatter plot of all the test samples with the color mapping corresponding to the predicted labels made by the decision tree classifier. The decision tree classifier follows that pattern that we saw previously of high retransmission and delayed acknowledgment counts correlates to a contaminated link. Using a 10 fold cross-validation gives a mean accuracy of 0.76, but the important thing to note is that it has a precision of 1 for the contaminated class. This is important if we use the layered approach because all the contaminated samples were labeled as contaminated which would then be verified

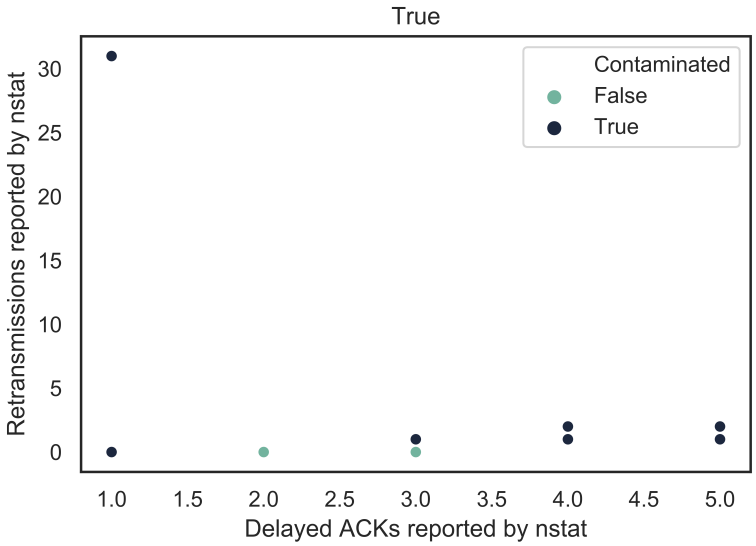
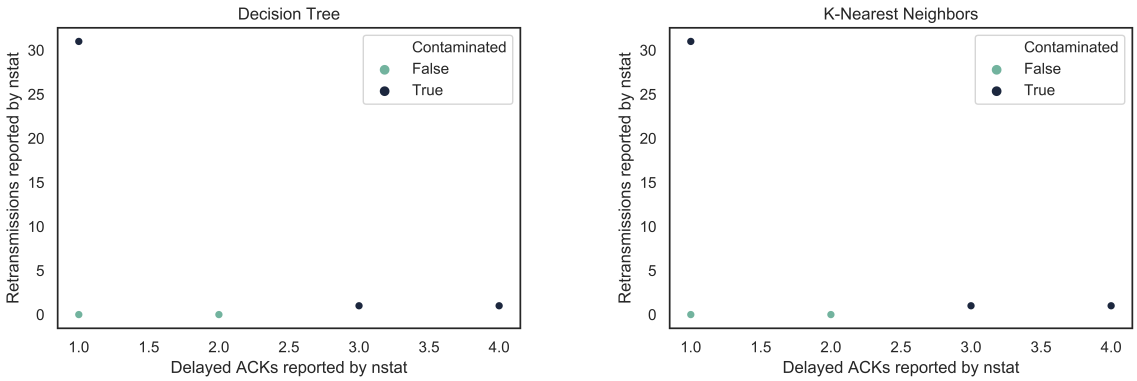


Figure 4.6: Scatter plot of data with true labels



a) Classification using a decision tree classifier

b) Classification using a k-nearest neighbors classifier

Figure 4.7: Classification using machine learning techniques

by looking at the power difference of the relevant links, even if not all of the clean samples were labeled clean.

Figure 4.7 (b) shows the scatter plot of all the test samples with the color mapping corresponding to the predicted labels made by the k-nearest neighbors classifier. The KNN classifier is eerily similar to the results of the decision tree classifier. It is classifying the results in the same way where higher retransmission and delayed acknowledgment counts

result in a predicted label of contaminated. It also has the desirable feature of having a precision of 1 for the contaminated labels. This classifier had an accuracy mean of 0.75 using a 10 fold cross-validation.

Both of these techniques can detect contaminated links decently using just the two most varying features, in fact adding the other features decreases the accuracy due to them having little to no variance. There is a reason that these are quite similar though, each sample is only recording the retransmissions throughout one minute. Many of the samples reported the number of delayed acknowledgments and retransmissions as 1 due to the sample period being so short meaning less time for retransmissions to happen. To explore this data set even further to see how these two features change over longer samples, the sum of both features over every 50 samples was computed then ran through the same data analysis as above.

Figure 4.8 is the scatter plot of the true labels after aggregating 50 samples into 1, going from 2000 samples down to just 40. This provides a much more clear picture of how the contamination is affecting the network performance over time which isn't so obvious when looking at the 1 minute samples.

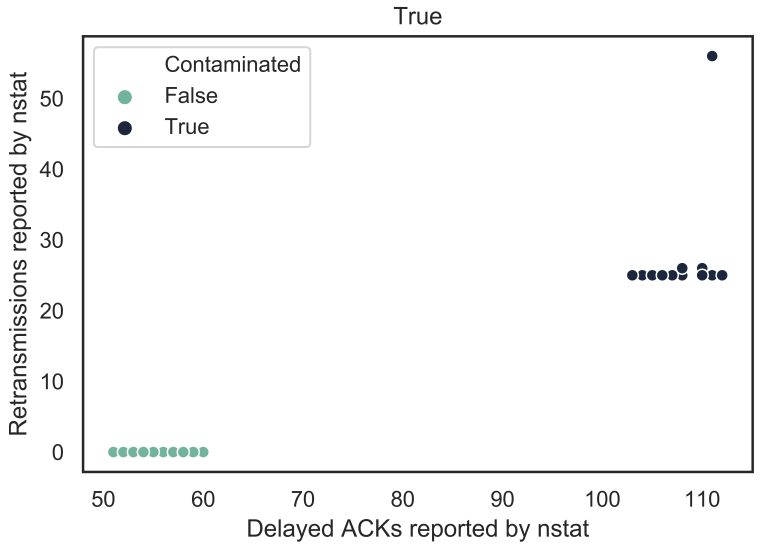
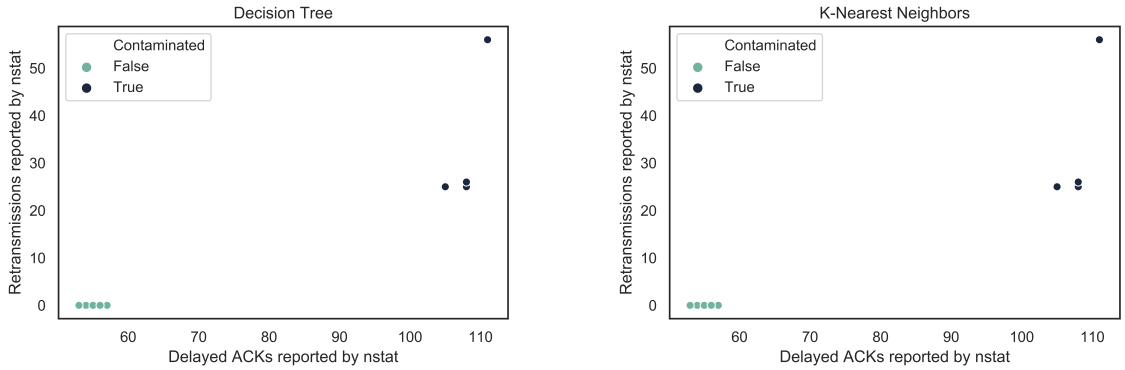


Figure 4.8: True labels after aggregation plotted



a) Classification using a decision tree classifier after aggregation

b) Classification using a k-nearest neighbors classifier after aggregation

Figure 4.9: Classification using machine learning techniques

Figure 4.9 (a) shows the results of the decision tree classifier on a scatter plot after aggregation. The aggregation boosted the performance of the classifier. It follows the same pattern that has been consistent throughout these experiments however now that is much clearer after the aggregation the decision tree classifier has an accuracy of 1.

Figure 4.9 (b) shows the results of the k-nearest neighbor classifier on a scatter plot after aggregation. This classifier also achieved an accuracy of 1 and is identical in performance to the decision tree classifier.

4.3.2 Second Analysis

However, one sample set was not enough to verify that it was the contamination that was causing these errors. The same methodology was used to run several more experiments. Some of the new experiments had the topology connected to the internet to see if this technique is even feasible on a simple network with the random factors that the web traffic provides. The results were inconsistent, the following experiments after the initial one did not show any significant change going from a clean to a contaminated connector. The data points were still run through the decision tree classifier and the KNN classifier with lackluster results. The accuracy for the decision tree was 0.52 and the accuracy for the KNN classifier was 0.51, at best.

4.3.3 Top-Down Approach Experiment Conclusions

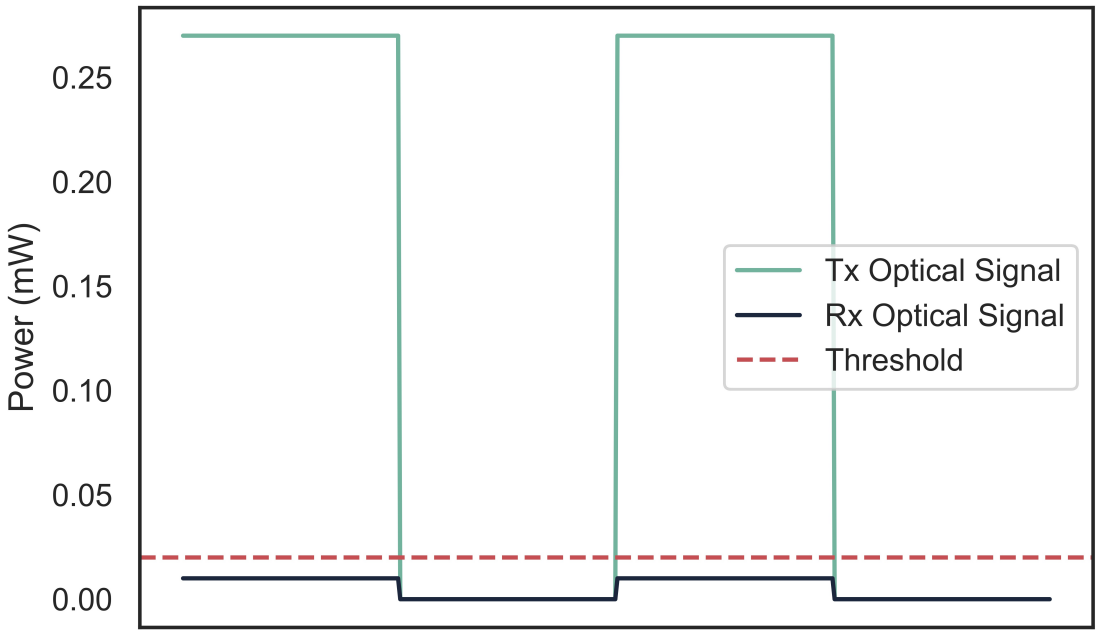


Figure 4.10: Static power loss causing loss of signal

The layered approach to detecting contaminated fibers was inconclusive, due to the inconsistencies of the higher layer protocol statistics. Although the initial results seemed promising it was not enough to be sure that it was the contaminated connectors causing the errors. Thus, analysis to explain this behavior was done. The connectors were contaminated to the point just before there was loss of signal, well below the receiver sensitivity (-17 dBm) at about -23 dBm and ping was used to monitor for corrupted bits. Even when the signal attenuation was almost at 100% there were still no corrupted bits that were seen. This presents a binary situation for these experiments, either the network performs well even with contamination or there is loss of signal due to contamination. Some insight is provided from the power experiments, in Figures 4.4 and 4.3 there was almost no variation

in the power lost over 24 hours meaning that the loss is static. As long as the power level for signaling a 1 is higher than the quantizer threshold then the network should function correctly, on the other hand, if the power level is lower than the threshold then, due to the static nature of the attenuation caused by contamination, all the 1s will end up being 0s causing loss of signal. This idea is illustrated in Figure 4.10.

Chapter 5

Conclusion

This work looked at the effects of contamination on optical fiber connectors and developed inequality 4.14 to detect contaminated optical connectors. The inequality detects contaminated fiber connectors through the use of Small Form-Factor Pluggable (SFP) Digital Optical Monitoring (DOM) power readings. Because the inequality leverages SFP power measurements to detect contamination with certainty, the tolerance range of the SFP power measurements had to be considered. [10] suggests a wide tolerance range of $\pm$ 3 dB which is much too wide to detect contamination that only affects the optical power slightly. To verify that the tolerance range is much smaller than that in practice the power readings from the SFPs and the power readings of a dedicated power meter were compared. All power losses due to cabling and connectors were considered when comparing these values and the results are shown in Table 4.1. Even taking the worst-case scenario for each of the comparisons the tolerance range of the SFP power measurements is no more than $\pm$ 1 dB, concluding that it is suitable to use these measurements for detection. Inequality 4.14 was successfully able to detect contaminated optical connectors during the experiments with no false positives. It is acknowledged that this inequality only holds true if the contamination affects optical power, however, if there is contamination present and the optical power is not changed then it should not cause instability or failure of the system. Some contamination could possibly cause thermal issues but no performance degradation but that is beyond the scope of this work. This work also discussed trying to detect contaminated fiber connectors by looking at statistics other than optical power. This included analyzing the exhaustive list of statistics shown in Appendix A using two machine learning techniques for classification. Although initially, the results seemed promising using a decision tree model

and K-Nearest Neighbors (KNN) model, ultimately it proved to be inconclusive and have poor performance after many trials. However, it did lead to the conclusion that without perturbation of the contaminant used, a static loss of optical power is seen and will result in either no bit errors or complete loss of signal.

In the future this work could be expanded by verifying that Inequality 4.14 will scale to much larger networks where it would prove to be more useful and finding more sophisticated methods to detect contaminated fiber connectors at higher levels of the OSI model using the list of statistics in Appendix A.

References

[1] A. Li, "Four common types of fiber optic connectors." `http://www.fiber-optic-solutions.com/four-common-types-of-fiber-optic-connectors.html`, Aug 2015. Accessed: 2018-10-26.

[2] N. Samaan and A. Karmouch, "Towards autonomic network management: an analysis of current and future research directions," *IEEE Communications Surveys Tutorials*, vol. 11, pp. 22–36, Aug 2009.

[3] H. Kim and N. Feamster, "Improving network management with software defined networking," *IEEE Communications Magazine*, vol. 51, pp. 114–119, February 2013.

[4] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: Enabling innovation in campus networks," *SIGCOMM Comput. Commun. Rev.*, vol. 38, pp. 69–74, Mar 2008.

[5] S. Lytle, M. Brown, T. Berdinskikh, D. Wilson, D. Fisher, *et al.*, "Correlation study between contamination and signal degradation in single-mode APC connectors," in *Photonics North 2009*, vol. 7386, pp. 73861W 1–10, International Society for Optics and Photonics, Aug 2009.

[6] J. L. Jewell, J. P. Harbison, A. Scherer, Y. H. Lee, and L. T. Florez, "Vertical-cavity surface-emitting lasers: Design, growth, fabrication, characterization," *IEEE Journal of Quantum Electronics*, vol. 27, pp. 1332–1346, June 1991.

[7] R. A. Morgan, "Vertical-cavity surface-emitting lasers: present and future," vol. 3003, pp. 3003 – 3003 – 13, Apr 1997.

[8] J. Stone and D. Marcuse, "Ultrahigh finesse fiber fabry-perot interferometers," *Journal of Lightwave Technology*, vol. 4, pp. 382–385, April 1986.

[9] B. Steindl, M. Hofbauer, K. Schneider-Hornstein, P. Brandl, and H. Zimmermann, "Single-photon avalanche photodiode based fiber optic receiver for up to 200 mb/s," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 24, pp. 1–8, March 2018.

[10] SFF Committee, "SFF-8472 diagnostic monitoring interface for optical transceivers." `http://www.10gtek.com/templates/wzten/pdf/SFF-8472-(Diagnostic\%20Monitoring\%20Interface).pdf`, Sept 2014. Accessed: 2018-10-26.

[11] "Small form-factor pluggable (SFP) transceiver multisource agreement (MSA)." `http://schelto.com/SFP/SFP\%20MSA.pdf`, Sept 2000. Accessed: 2018-10-26.

[12] IEEE, "IEEE standard for ethernet," *IEEE Std 802.3-2018 (Revision of IEEE Std 802.3-2015)*, pp. 1–5600, Aug 2018.

[13] N. Albeanu, L. Aseere, T. Berdinskikh, , J. Nguyen, Y. Pradieu, D. Silmser, H. Tkalec, and E. Tse, "Optical connector contamination and it's influence on optical signal performance," tech. rep., Celestica International, Nov 2017.

[14] M. Kihara, M. Okada, M. Hosoda, T. Iwata, Y. Yajima, and M. Toyonaga, "Tool for inspecting faults from incorrectly cleaved fiber ends and contaminated optical fiber connector end surfaces," *Optical Fiber Technology*, vol. 18, pp. 470 – 479, Aug 2012.

[15] T. Berdinskikh, A. Ho, J. Garcia, C. Gleason, S.-Y. Huang, J. Kilmer, S. Lytle, T. Mitcheltree, B. J. Roche, H. Tkalec, D. H. Wilson, and F. Zhang, "Development of cleanliness specifications for single-mode connectors with 1.25 and 2.5 mm ferrules," in *2006 Optical Fiber Communication Conference and the National Fiber Optic Engineers Conference*, pp. 1–10, Mar 2006.

[16] T. Berdinskikh, J. Bragg, E. Tse, J. Daniel, P. Arrowsmith, A. Fisenko, and S. Mahmoud, "The contamination of fiber optics connectors and their effect on optical performance," in *Optical Fiber Communication Conference and Exhibit*, pp. 617–619, Mar 2002.

[17] S. Takahashi, Y. Sadohara, C. Gleason, T. Berdinskikh, T. Mitcheltree, and S. Lytle, "Contamination influence on receptacle type optical data links," in *Photonic Applications in Devices and Communication Systems*, vol. 5970, p. 597004, International Society for Optics and Photonics, Sept 2005.

[18] C. Chen, Y. Sadohara, T. Berdinskikh, D. Fisher, M. Brown, B. Roche, B. Coviello, and D. Wilson, "Contamination effects on optical performance for short reach 10Gb/s SFP+ transceivers," in *National Fiber Optic Engineers Conference*, pp. JThA56 1–3, Optical Society of America, Mar 2010.

[19] G. Proudly and H. White, "Contamination effects in single-mode optical fiber connectors," in *Photonic Applications for Aerospace, Commercial, and Harsh Environments IV*, vol. 8720, pp. 87200W 1–9, May 2013.

[20] T. Berdinskikh, A. I. Fisenko, J. Daniel, J. Bragg, and D. Phillips, "The role of electrostatic charge effect on the contamination of fiber optics connectors and the ways of eliminating it," in *Applications of Photonic Technology 5*, vol. 4833, pp. 420–432, International Society for Optics and Photonics, Feb 2003.

[21] Z. He, W. Mahmood, E. Sahinci, N. Ahmad, and Y. Pradieu, "Analysis on the effects of fiber end face scratches on return loss performance of optical fiber connectors," *Journal of Lightwave Technology*, vol. 22, pp. 2749–2754, Dec 2004.

[22] C. Fukai, M. Kihara, R. Koyama, K. Saito, Y. Abe, T. Kurashima, and K. Katayama, "Investigation of connector performance and damage to optical fiber connectors with dust during high power transmission," *Journal of Lightwave Technology*, pp. 1–1, June 2018.

[23] A. Patcha and J.-M. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends," *Computer Networks*, vol. 51, pp. 3448 – 3470, Aug 2007.

[24] M. Thottan and C. Ji, "Anomaly detection in IP networks," *IEEE Transactions on Signal Processing*, vol. 51, pp. 2191–2204, Aug 2003.

[25] S. A. Mehdi, J. Khalid, and S. A. Khayam, "Revisiting traffic anomaly detection using software defined networking," in *Recent Advances in Intrusion Detection*, vol. 14, pp. 161–180, Sept 2011.

[26] A. Kulkarni, Y. Pino, M. French, and T. Mohsenin, "Real-time anomaly detection framework for many-core router through machine-learning techniques," *J. Emerg. Technol. Comput. Syst.*, vol. 13, pp. 10:1–10:22, June 2016.

[27] B. Eriksson, P. Barford, R. Bowden, N. Duffield, J. Sommers, and M. Roughan, "Basisdetect: A model-based network event detection framework," in *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement*, pp. 451–464, Nov 2010.

[28] S. Kandula, D. Katabi, and J.-P. Vasseur, "Shrink: A tool for failure diagnosis in ip networks," in *Proceedings of the 2005 ACM SIGCOMM Workshop on Mining Network Data*, pp. 173–178, Aug 2005.

[29] M. Ruiz, F. Fresi, A. P. Vela, G. Meloni, N. Sambo, F. Cugini, L. Poti, L. Velasco, and P. Castoldi, "Service-triggered failure identification/localization through monitoring of multiple parameters," in *ECOC 2016; 42nd European Conference on Optical Communication*, pp. 1–3, Sept 2016.

[30] A. P. Vela, M. Ruiz, F. Fresi, N. Sambo, F. Cugini, G. Meloni, L. Pot, L. Velasco, and P. Castoldi, "BER degradation detection and failure identification in elastic optical networks," *Journal of Lightwave Technology*, vol. 35, pp. 4595–4604, Nov 2017.

[31] C. Mas and P. Thiran, "An efficient algorithm for locating soft and hard failures in WDM networks," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 1900–1911, Oct 2000.

[32] C. Manikopoulos and S. Papavassiliou, "Network intrusion and fault detection: a statistical anomaly approach," *IEEE Communications Magazine*, vol. 40, pp. 76–82, Oct 2002.

[33] E. Dart, L. Rotman, B. Tierney, M. Hester, and J. Zurawski, "The Science DMZ: A network design pattern for data-intensive science," in *2013 SC - International Conference for High Performance Computing, Networking, Storage and Analysis (SC)*, pp. 1–10, Nov 2013.

[34] A. Bhoite, N. Beke, T. Duffy, M. Moore, and M. Torres, "Automated fiber optic cable endface field inspection technology," in *2011 IEEE AUTOTESTCON*, pp. 226–234, IEEE, Sept 2011.

Appendix A

Network Data

The following appendix is a compiled list of network data statistics that can be obtained to analyze the performance of the network.

A.1 Layer 1 Statistics Tables

Table A.1: Layer 1 Statistics Description

Data Element	**Vantage Point**	**Collection Method**
Supply Voltage	Switch	SSH, Possibly SNMP
Teperature	Switch	SSH, Possibly SNMP
Rx Power	Switch	SSH, Possibly SNMP
Tx Power	Switch	SSH, Possibly SNMP
Laser Bias	Switch	SSH, Possibly SNMP
Tx Fault	Switch	SSH, Possibly SNMP
Loss of Signal	Switch	SSH, Possibly SNMP
Link Length	Switch	SSH, Possibly SNMP

Table A.2: Layer 1 Statistics Description

Data Element	**Stat Description**
Supply Voltage	Voltage driving SFP

Continued on next page

Table A.2 – *Continued from previous page*

Data Element	**Stat Description**
Teperature	Temperature of SFP
Rx Power	Rx Optical Power
Tx Power	Tx Optical Power
Laser Bias	Bias current of the laser
Tx Fault	Faults with transmitting data
Loss of Signal	Low or no power causing a loss of signal
Link Length	Test the length of a link

A.2 Layer 2 Statistics Tables

Table A.3: Layer 2 Statistics Description

Data Element	**Vantage Point**	**Collection Method**
Total Packets Received (Octets)	Switch	SSH
Packets Received 64 Octets	Switch	SSH
Packets Received 65–127 Octets	Switch	SSH
Packets Received 128–255 Octets	Switch	SSH
Packets Received 256–511 Octets	Switch	SSH
Packets Received 512–1023 Octets	Switch	SSH
Packets Received 1024–1518 Octets	Switch	SSH
Packets Received >1518 Octets	Switch	SSH
Packets RX and TX 64 Octets	Switch	SSH
Packets RX and TX 65–127 Octets	Switch	SSH
Packets RX and TX 128–255 Octets	Switch	SSH
Packets RX and TX 256–511 Octets	Switch	SSH

Continued on next page

Table A.3 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
Packets RX and TX 512–1023 Octets	Switch	SSH
Packets RX and TX 1024–1518 Octets	Switch	SSH
Packets RX and TX 1519–2047 Octets	Switch	SSH
Packets RX and TX 2048–4095 Octets	Switch	SSH
Packets RX and TX 4096–9216 Octets	Switch	SSH
Total Packets Received Without Error	Switch	SSH
Unicast Packets Received	Switch	SSH
Multicast Packets Received	Switch	SSH
Broadcast Packets Received	Switch	SSH
Total Packets Received with MAC Errors	Switch	SSH
Jabbers Received	Switch	SSH
Fragments/Undersize Received	Switch	SSH
Alignment Errors	Switch	SSH
FCS Errors	Switch	SSH
Overruns	Switch	SSH
Total Received Packets Not Forwarded	Switch	SSH
802.3x Pause Frames Received	Switch	SSH
Unacceptable Frame Type	Switch	SSH
Total Packets Transmitted (Octets)	Switch	SSH
Packets Transmitted 64 Octets	Switch	SSH
Packets Transmitted 65–127 Octets	Switch	SSH
Packets Transmitted 128–255 Octets	Switch	SSH
Packets Transmitted 256–511 Octets	Switch	SSH
Packets Transmitted 512–1023 Octets	Switch	SSH
Packets Transmitted 1024–1518 Octets	Switch	SSH

Continued on next page

Table A.3 – *Continued from previous page*

Data Element	**Vantage Point**	**Collection Method**
Packets Transmitted >1518 Octets	Switch	SSH
Max Frame Size	Switch	SSH
Total Packets Transmitted Successfully	Switch	SSH
Unicast Packets Transmitted	Switch	SSH
Multicast Packets Transmitted	Switch	SSH
Broadcast Packets Transmitted	Switch	SSH
Total Transmit Errors	Switch	SSH
Total Transmit Packets Discards	Switch	SSH
Single Collision Frames	Switch	SSH
Multiple Collision Frames	Switch	SSH
Excessive Collisions	Switch	SSH
802.3x Pause Frames Transmitted	Switch	SSH
GVRP PDUs Received	Switch	SSH
GVRP PDUs Transmitted	Switch	SSH
GVRP Failed Registrations	Switch	SSH
GMRP PDUs Received	Switch	SSH
GMRP PDUs Transmitted	Switch	SSH
GMRP Failed Registrations	Switch	SSH
STP BPDUs Transmitted	Switch	SSH
STP BPDUs Received	Switch	SSH
RST BPDUs Transmitted	Switch	SSH
RSTP BPDUs Received	Switch	SSH
MSTP BPDUs Transmitted	Switch	SSH
MSTP BPDUs Received	Switch	SSH
EAPOL Frames Transmitted	Switch	SSH

Continued on next page

Table A.3 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
EAPOL Frames Received	Switch	SSH
Time Since Counters Last Cleared	Switch	SSH
Rx packets	Linux Host	ifconig
Rx errors	Linux Host	ifconig
Rx drops	Linux Host	ifconig
Rx overruns	Linux Host	ifconig
Rx frame	Linux Host	ifconig
Tx packets	Linux Host	ifconig
Tx errors	Linux Host	ifconig
Tx drops	Linux Host	ifconig
Tx overruns	Linux Host	ifconig
Tx carrier	Linux Host	ifconig

Table A.4: Layer 2 Statistics Description

Data Element	Stat Description
Total Packets Received (Octets)	Total Bytes Received by Switch
Packets Received 64 Octets	Packets Received with size of exactly 64 bytes
Packets Received 65–127 Octets	Packets Received with size between 65–127 bytes
Packets Received 128–255 Octets	Packets Received with size between 128–255 bytes
Packets Received 256–511 Octets	Packets Received with size between 256–511 bytes

Continued on next page

Table A.4 – *Continued from previous page*

Data Element	Stat Description
Packets Received 512–1023 Octets	Packets Received with size between 512–1023 bytes
Packets Received 1024–1518 Octets	Packets Received with size between 1024–1518 bytes
Packets Received >1518 Octets	Packets Received with size greater than 1518 bytes
Packets RX and TX 64 Octets	Packets Sent or Received with size of exactly 64 bytes
Packets RX and TX 65–127 Octets	Packets Sent or Received with size between 65–127 bytes
Packets RX and TX 128–255 Octets	Packets Sent or Received with size between 128–255 bytes
Packets RX and TX 256–511 Octets	Packets Sent or Received with size between 256–511 bytes
Packets RX and TX 512–1023 Octets	Packets Sent or Received with size between 512–1023 bytes
Packets RX and TX 1024–1518 Octets	Packets Sent or Received with size between 1024–1518 bytes
Packets RX and TX 1519–2047 Octets	Packets Sent or Received with size between 1519–2047 bytes
Packets RX and TX 2048–4095 Octets	Packets Sent or Received with size between 2048–4095 bytes
Packets RX and TX 4096–9216 Octets	Packets Sent or Received with size between 4096–9216 bytes

Continued on next page

Table A.4 – *Continued from previous page*

Data Element	**Stat Description**
Total Packets Received Without Error	Total Packets Received with no error but could have still been dropped due to other issues such as a full buffer
Unicast Packets Received	Packets Received with only one destination
Multicast Packets Received	Packets Received with more than one destination
Broadcast Packets Received	Broadcast Packets Received
Total Packets Received with MAC Errors	Packets Received with Ethernet Errors
Jabbers Received	The total number of packets received that were longer than 1518 octets (excluding framing bits, but including FCS octets), and had either a bad Frame Check Sequence (FCS) with an integral number of octets (FCS Error) or a bad FCS with a non–integral number of octets (Alignment Error).
Fragments/Undersize Received	Packets received with length smaller than 64 bytes.
Alignment Errors	Frames that either have a bad FCS or have an uneven number of bits
FCS Errors	Bad Frame Check Sequence (FCS)
Overruns	When the buffer gets full and has to drop packets
Total Received Packets Not Forwarded	A count of valid frames received which were discarded (in other words, filtered) by the forwarding process

Continued on next page

Table A.4 – *Continued from previous page*

Data Element	Stat Description
802.3x Pause Frames Received	A count of MAC Control frames received on this interface with an opcode indicating the PAUSE operation. This counter does not increment when the interface is operating in half–duplex mode.
Unacceptable Frame Type	Bad Frame type, not ethernet.
Total Packets Transmitted (Octets)	Total bytes transmitted
Packets Transmitted 64 Octets	Packets Sent with size of exactly 64 bytes
Packets Transmitted 65–127 Octets	Packets sent with size between 65–127 bytes
Packets Transmitted 128–255 Octets	Packets sent with size between 128–255 bytes
Packets Transmitted 256–511 Octets	Packets sent with size between 256–511 bytes
Packets Transmitted 512–1023 Octets	Packets sent with size between 512–1023 bytes
Packets Transmitted 1024–1518 Octets	Packets sent with size between 1024–1518 bytes
Packets Transmitted >1518 Octets	Packets sent with size greater than 1518 bytes
Max Frame Size	Maximum ethernet frame size, not including MAC information.
Total Packets Transmitted Successfully	The number of frames that have been transmitted by this port to its segment.
Unicast Packets Transmitted	The total number of packets that higher–level protocols requested be transmitted to a subnetwork–unicast address, including those that were discarded or not sent.

Continued on next page

Table A.4 – *Continued from previous page*

Data Element	**Stat Description**
Multicast Packets Transmitted	The total number of packets that higher–level protocols requested be transmitted to a Multicast address, including those that were discarded or not sent.
Broadcast Packets Transmitted	The total number of packets that higher–level protocols requested be transmitted to the Broadcast address, including those that were discarded or not sent.
Total Transmit Errors	The sum of Single, Multiple, and Excessive Collisions.
Total Transmit Packets Discards	The sum of single collision frames discarded, multiple collision frames discarded, and excessive frames discarded.
Single Collision Frames	A count of the number of successfully transmitted frames on a particular interface for which transmission is inhibited by exactly one collision.
Multiple Collision Frames	A count of the number of successfully transmitted frames on a particular interface for which transmission is inhibited by more than one collision.
Excessive Collisions	A count of frames for which transmission on a particular interface fails due to excessive collisions.

Continued on next page

Table A.4 – *Continued from previous page*

Data Element	Stat Description
802.3x Pause Frames Transmitted	A count of MAC Control frames transmitted on this interface with an opcode indicating the PAUSE operation. This counter does not increment when the interface is operating in half–duplex mode.
GVRP PDUs Received	The count of GVRP PDUs received in the GARP layer.
GVRP PDUs Transmitted	The count of GVRP PDUs transmitted from the GARP layer.
GVRP Failed Registrations	The number of times attempted GVRP registrations could not be completed.
GMRP PDUs Received	The count of GMRP PDUs received in the GARP layer.
GMRP PDUs Transmitted	The count of GMRP PDUs transmitted from the GARP layer.
GMRP Failed Registrations	The number of times attempted GMRP registrations could not be completed.
STP BPDUs Transmitted	Spanning Tree Protocol Bridge Protocol Data Units sent.
STP BPDUs Received	Spanning Tree Protocol Bridge Protocol Data Units received.
RST BPDUs Transmitted	Rapid Spanning Tree Protocol Bridge Protocol Data Units sent.
RSTP BPDUs Received	Rapid Spanning Tree Protocol Bridge Protocol Data Units received.

Continued on next page

Table A.4 – *Continued from previous page*

Data Element	**Stat Description**
MSTP BPDUs Transmitted	Multiple Spanning Tree Protocol Bridge Protocol Data Units sent.
MSTP BPDUs Received	Multiple Spanning Tree Protocol Bridge Protocol Data Units received
EAPOL Frames Transmitted	The number of EAPOL frames of any type that have been transmitted by this authenticator.
EAPOL Frames Received	The number of valid EAPOL frames of any type that have been received by this authenticator.
Time Since Counters Last Cleared	Time since the switch counters were last cleared
Rx packets	Number of Packets Received by an interface
Rx errors	Number of received errored packets received by an interface
Rx drops	Number of received packets dropped by an interface
Rx overruns	When the receive buffer gets full and has to drop packets at an interface
Rx frame	When a received packets has an FCS error at an interface
Tx packets	Number of packets transmitted by an interface
Tx errors	Number of packets that were to be transmitted that have an error at an interface

Continued on next page

Table A.4 – *Continued from previous page*

Data Element	Stat Description
Tx drops	Number of packets that were to be transmitted are dropped at an interface
Tx overruns	Packets that are dropped due to a full transmit buffer at an interface
Tx carrier	Collisions of transmitted packets from an interface

A.3 Layer 3 Statistics Tables

Table A.5: Layer 3 Statistics Description

Data Element	Vantage Point	Collection Method
IpInReceives	Switch	SSH, Possibly SNMP
IpInHdrErrors	Switch	SSH, Possibly SNMP
IpInAddrErrors	Switch	SSH, Possibly SNMP
IpForwDatagrams	Switch	SSH, Possibly SNMP
IpInUnknownProtos	Switch	SSH, Possibly SNMP
IpInDiscards	Switch	SSH, Possibly SNMP
IpInDelivers	Switch	SSH, Possibly SNMP
IpOutRequests	Switch	SSH, Possibly SNMP
IpOutDiscards	Switch	SSH, Possibly SNMP
IpOutNoRoutes	Switch	SSH, Possibly SNMP
IpReasmTimeout	Switch	SSH, Possibly SNMP
IpReasmReqds	Switch	SSH, Possibly SNMP
IpReasmOKs	Switch	SSH, Possibly SNMP

Continued on next page

Table A.5 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
IpReasmFails	Switch	SSH, Possibly SNMP
IpFragOKs	Switch	SSH, Possibly SNMP
IpFragFails	Switch	SSH, Possibly SNMP
IpFragCreates	Switch	SSH, Possibly SNMP
IpRoutingDiscards	Switch	SSH, Possibly SNMP
IpInReceives	Linux Host	nstat
IpInHdrErrors	Linux Host	nstat
IpInAddrErrors	Linux Host	nstat
IpForwDatagrams	Linux Host	nstat
IpInUnknownProtos	Linux Host	nstat
IpInDiscards	Linux Host	nstat
IpInDelivers	Linux Host	nstat
IpOutRequests	Linux Host	nstat
IpOutDiscards	Linux Host	nstat
IpOutNoRoutes	Linux Host	nstat
IpReasmTimeout	Linux Host	nstat
IpReasmReqds	Linux Host	nstat
IpReasmOKs	Linux Host	nstat
IpReasmFails	Linux Host	nstat
IpFragOKs	Linux Host	nstat
IpFragFails	Linux Host	nstat
IpFragCreates	Linux Host	nstat
Ip6InReceives	Linux Host	nstat
Ip6InHdrErrors	Linux Host	nstat
Ip6InTooBigErrors	Linux Host	nstat

Continued on next page

Table A.5 – *Continued from previous page*

Data Element	**Vantage Point**	**Collection Method**
Ip6InNoRoutes	Linux Host	nstat
Ip6InAddrErrors	Linux Host	nstat
Ip6InUnknownProtos	Linux Host	nstat
Ip6InTruncatedPkts	Linux Host	nstat
Ip6InDiscards	Linux Host	nstat
Ip6InDelivers	Linux Host	nstat
Ip6OutForwDatagrams	Linux Host	nstat
Ip6OutRequests	Linux Host	nstat
Ip6OutDiscards	Linux Host	nstat
Ip6OutNoRoutes	Linux Host	nstat
Ip6ReasmTimeout	Linux Host	nstat
Ip6ReasmReqds	Linux Host	nstat
Ip6ReasmOKs	Linux Host	nstat
Ip6ReasmFails	Linux Host	nstat
Ip6FragOKs	Linux Host	nstat
Ip6FragFails	Linux Host	nstat
Ip6FragCreates	Linux Host	nstat
Ip6InMcastPkts	Linux Host	nstat
Ip6OutMcastPkts	Linux Host	nstat
Ip6InOctets	Linux Host	nstat
Ip6OutOctets	Linux Host	nstat
Ip6InMcastOctets	Linux Host	nstat
Ip6OutMcastOctets	Linux Host	nstat
Ip6InBcastOctets	Linux Host	nstat
Ip6OutBcastOctets	Linux Host	nstat

Continued on next page

Table A.5 – *Continued from previous page*

Data Element	**Vantage Point**	**Collection Method**
Ip6InNoECTPkts	Linux Host	nstat
Ip6InECT1Pkts	Linux Host	nstat
Ip6InECT0Pkts	Linux Host	nstat
Ip6InCEPkts	Linux Host	nstat
IpExtInNoRoutes	Linux Host	nstat
IpExtInTruncatedPkts	Linux Host	nstat
IpExtInMcastPkts	Linux Host	nstat
IpExtOutMcastPkts	Linux Host	nstat
IpExtInBcastPkts	Linux Host	nstat
IpExtOutBcastPkts	Linux Host	nstat
IpExtInOctets	Linux Host	nstat
IpExtOutOctets	Linux Host	nstat
IpExtInMcastOctets	Linux Host	nstat
IpExtOutMcastOctets	Linux Host	nstat
IpExtInBcastOctets	Linux Host	nstat
IpExtOutBcastOctets	Linux Host	nstat
IpExtInCsumErrors	Linux Host	nstat
IpExtInNoECTPkts	Linux Host	nstat
IpExtInECT1Pkts	Linux Host	nstat
IpExtInECT0Pkts	Linux Host	nstat
IpExtInCEPkts	Linux Host	nstat

Table A.6: Layer 3 Statistics Description

Data Element	Stat Description
IpInReceives	IP packets received
IpInHdrErrors	IP packets received with header errors
IpInAddrErrors	IP packets received with an invalid address
IpForwDatagrams	Datagrams that have been forwarded through the switch
IpInUnknownProtos	IP packets received with an unknown or unsupported protocol
IpInDiscards	IP packets that have been discarded due to a full buffer
IpInDelivers	IP packets delivered to IP user-protocols such as ICMP
IpOutRequests	IP packets supplied by local IP-user protocols
IpOutDiscards	IP packets discarded due to a full buffer
IpOutNoRoutes	Packets discarded due to having no path on record to the destination
IpReasmTimeout	Fragmented packets that did not reassemble in time
IpReasmReqds	Fragmented packets that need to be reassembled at this interface
IpReasmOKs	Fragmented packets that have been reassembled
IpReasmFails	Fragmented packets that could not be reassembled

Continued on next page

Table A.6 – *Continued from previous page*

Data Element	**Stat Description**
IpFragOKs	Packets that have been successfully fragmented
IpFragFails	Packets that were not able to be fragmented
IpFragCreates	Number of fragments that have been created
IpRoutingDiscards	Routing entries that were discarded even if they were valid
IpInReceives	IP packets received
IpInHdrErrors	IP packets received with header errors
IpInAddrErrors	IP packets received with an invalid address
IpForwDatagrams	Datagrams that have been forwarded through this host
IpInUnknownProtos	IP packets received with an unknown or unsupported protocol
IpInDiscards	IP packets that have been discarded due to a full buffer
IpInDelivers	IP packets delivered to IP user-protocols such as ICMP
IpOutRequests	IP packets supplied by local IP-user protocols
IpOutDiscards	IP packets discarded due to a full buffer
IpOutNoRoutes	Packets discarded due to having no path on record to the destination
IpReasmTimeout	Fragmented packets that did not reassemble in time

Continued on next page

Table A.6 – *Continued from previous page*

Data Element	**Stat Description**
IpReasmReqds	Fragmented packets that need to be reassembled at this interface
IpReasmOKs	Fragmented packets that have been reassembled
IpReasmFails	Fragmented packets that could not be reassembled
IpFragOKs	Packets that have been successfully fragmented
IpFragFails	Packets that were not able to be fragmented
IpFragCreates	Number of fragments that have been created
Ip6InReceives	IPv6 packets received
Ip6InHdrErrors	IPv6 packets received with header errors
Ip6InTooBigErrors	Packets dropped due to size being greater than the MTU
Ip6InNoRoutes	IPv6 packets discarded due to having no path on record to the destination
Ip6InAddrErrors	IPv6 packets received with an invalid address
Ip6InUnknownProtos	IPv6 packets received with an unknown or unsupported protocol
Ip6InTruncatedPkts	IPv6 packets that have been truncated
Ip6InDiscards	IPv6 packets that have been discarded due to a full buffer
Ip6InDelivers	IPv6 packets delivered to IP user-protocols such as ICMP

Continued on next page

Table A.6 – *Continued from previous page*

Data Element	**Stat Description**
Ip6OutForwDatagrams	IPv6 datagrams that have been forwarded through this host
Ip6OutRequests	IPv6 packets supplied by local IP-user protocols
Ip6OutDiscards	IPv6 packets discarded due to a full buffer
Ip6OutNoRoutes	IPv6 packets discarded due to having no path on record to the destination
Ip6ReasmTimeout	Fragmented IPv6 packets that did not reassemble in time
Ip6ReasmReqds	Fragmented IPv6 packets that need to be reassembled at this interface
Ip6ReasmOKs	Fragmented IPv6 packets that have been reassembled
Ip6ReasmFails	Fragmented IPv6 packets that could not be reassembled
Ip6FragOKs	IPv6 packets that have been successfully fragmented
Ip6FragFails	IPv6 packets that were not able to be fragmented
Ip6FragCreates	Number of IPv6 fragments that have been created
Ip6InMcastPkts	Multicast packets received
Ip6OutMcastPkts	Multicast packets sent
Ip6InOctets	Bytes received by IPv6 address
Ip6OutOctets	Bytes sent by IPv6 address

Continued on next page

Table A.6 – *Continued from previous page*

Data Element	**Stat Description**
Ip6InMcastOctets	Multicast bytes received
Ip6OutMcastOctets	Multicast bytes sent
Ip6InBcastOctets	Broadcast bytes received
Ip6OutBcastOctets	Broadcast bytes sent
Ip6InNoECTPkts	Packets received with no congestion notification
Ip6InECT1Pkts	Packets received with a congestion notification of 1
Ip6InECT0Pkts	Packets received with a congestion notification of 0
Ip6InCEPkts	Packets received with congestion experienced
IpExtInNoRoutes	Packets received with no valid route
IpExtInTruncatedPkts	Received truncated packets
IpExtInMcastPkts	Received multicast packets
IpExtOutMcastPkts	Sent multicast packets
IpExtInBcastPkts	Received broadcast packets
IpExtOutBcastPkts	Sent broadcast packets
IpExtInOctets	Bytes received in IP packets
IpExtOutOctets	Bytes sent in IP packets
IpExtInMcastOctets	Bytes received from multicast packets
IpExtOutMcastOctets	Bytes sent from multicast packets
IpExtInBcastOctets	Bytes received from broadcast packets
IpExtOutBcastOctets	Bytes sent from broadcast packets
IpExtInCsumErrors	Packets received with a bad checksum

Continued on next page

Table A.6 – *Continued from previous page*

Data Element	Stat Description
IpExtInNoECTPkts	Packets received with no congestion notification
IpExtInECT1Pkts	Packets received with a congestion notification of 1
IpExtInECT0Pkts	Packets received with a congestion notification of 0
IpExtInCEPkts	Packets received with congestion experienced

A.4 Layer 4 Statistics Tables

Table A.7: Layer 4 Statistics Description

Data Element	Vantage Point	Collection Method
IcmpInMsgs	Switch	SSH, Possibly SNMP
IcmpInErrors	Switch	SSH, Possibly SNMP
IcmpInDestUnreachs	Switch	SSH, Possibly SNMP
IcmpInTimeExcds	Switch	SSH, Possibly SNMP
IcmpInParmProbs	Switch	SSH, Possibly SNMP
IcmpInSrcQuenchs	Switch	SSH, Possibly SNMP
IcmpInRedirects	Switch	SSH, Possibly SNMP
IcmpInEchos	Switch	SSH, Possibly SNMP
IcmpInEchoReps	Switch	SSH, Possibly SNMP
IcmpInTimestamps	Switch	SSH, Possibly SNMP
IcmpInTimestampReps	Switch	SSH, Possibly SNMP
IcmpInAddrMasks	Switch	SSH, Possibly SNMP

Continued on next page

Table A.7 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
IcmpInAddrMaskReps	Switch	SSH, Possibly SNMP
IcmpOutMsgs	Switch	SSH, Possibly SNMP
IcmpOutErrors	Switch	SSH, Possibly SNMP
IcmpOutDestUnreachs	Switch	SSH, Possibly SNMP
IcmpOutTimeExcds	Switch	SSH, Possibly SNMP
IcmpOutParmProbs	Switch	SSH, Possibly SNMP
IcmpOutSrcQuenchs	Switch	SSH, Possibly SNMP
IcmpOutRedirects	Switch	SSH, Possibly SNMP
IcmpOutEchos	Switch	SSH, Possibly SNMP
IcmpOutEchoReps	Switch	SSH, Possibly SNMP
IcmpOutTimestamps	Switch	SSH, Possibly SNMP
IcmpOutTimestampReps	Switch	SSH, Possibly SNMP
IcmpOutAddrMasks	Switch	SSH, Possibly SNMP
IcmpInMsgs	Linux Host	nstat
IcmpInErrors	Linux Host	nstat
IcmpInCsumErrors	Linux Host	nstat
IcmpInDestUnreachs	Linux Host	nstat
IcmpInTimeExcds	Linux Host	nstat
IcmpInParmProbs	Linux Host	nstat
IcmpInSrcQuenchs	Linux Host	nstat
IcmpInRedirects	Linux Host	nstat
IcmpInEchos	Linux Host	nstat
IcmpInEchoReps	Linux Host	nstat
IcmpInTimestamps	Linux Host	nstat
IcmpInTimestampReps	Linux Host	nstat

Continued on next page

Table A.7 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
IcmpInAddrMasks	Linux Host	nstat
IcmpInAddrMaskReps	Linux Host	nstat
IcmpOutMsgs	Linux Host	nstat
IcmpOutErrors	Linux Host	nstat
IcmpOutDestUnreachs	Linux Host	nstat
IcmpOutTimeExcds	Linux Host	nstat
IcmpOutParmProbs	Linux Host	nstat
IcmpOutSrcQuenchs	Linux Host	nstat
IcmpOutRedirects	Linux Host	nstat
IcmpOutEchos	Linux Host	nstat
IcmpOutEchoReps	Linux Host	nstat
IcmpOutTimestamps	Linux Host	nstat
IcmpOutTimestampReps	Linux Host	nstat
IcmpOutAddrMasks	Linux Host	nstat
IcmpOutAddrMaskReps	Linux Host	nstat
IcmpMsgInType3	Linux Host	nstat
IcmpMsgInType11	Linux Host	nstat
IcmpMsgOutType3	Linux Host	nstat
IcmpMsgOutType11	Linux Host	nstat
TcpActiveOpens	Linux Host	nstat
TcpPassiveOpens	Linux Host	nstat
TcpAttemptFails	Linux Host	nstat
TcpEstabResets	Linux Host	nstat
TcpInSegs	Linux Host	nstat
TcpOutSegs	Linux Host	nstat

Continued on next page

Table A.7 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
TcpRetransSegs	Linux Host	nstat
TcpInErrs	Linux Host	nstat
TcpOutRsts	Linux Host	nstat
TcpInCsumErrors	Linux Host	nstat
UdpInDatagrams	Linux Host	nstat
UdpNoPorts	Linux Host	nstat
UdpInErrors	Linux Host	nstat
UdpOutDatagrams	Linux Host	nstat
UdpRcvbufErrors	Linux Host	nstat
UdpSndbufErrors	Linux Host	nstat
UdpInCsumErrors	Linux Host	nstat
UdpIgnoredMulti	Linux Host	nstat
UdpLiteInDatagrams	Linux Host	nstat
UdpLiteNoPorts	Linux Host	nstat
UdpLiteInErrors	Linux Host	nstat
UdpLiteOutDatagrams	Linux Host	nstat
UdpLiteRcvbufErrors	Linux Host	nstat
UdpLiteSndbufErrors	Linux Host	nstat
UdpLiteInCsumErrors	Linux Host	nstat
UdpLiteIgnoredMulti	Linux Host	nstat
Icmp6InMsgs	Linux Host	nstat
Icmp6InErrors	Linux Host	nstat
Icmp6OutMsgs	Linux Host	nstat
Icmp6OutErrors	Linux Host	nstat
Icmp6InCsumErrors	Linux Host	nstat

Continued on next page

Table A.7 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
Icmp6InDestUnreachs	Linux Host	nstat
Icmp6InPktTooBigs	Linux Host	nstat
Icmp6InTimeExcds	Linux Host	nstat
Icmp6InParmProblems	Linux Host	nstat
Icmp6InEchos	Linux Host	nstat
Icmp6InEchoReplies	Linux Host	nstat
Icmp6InGroupMembQueries	Linux Host	nstat
Icmp6InGroupMembResponses	Linux Host	nstat
Icmp6InGroupMembReductions	Linux Host	nstat
Icmp6InRouterSolicits	Linux Host	nstat
Icmp6InRouterAdvertisements	Linux Host	nstat
Icmp6InNeighborSolicits	Linux Host	nstat
Icmp6InNeighborAdvertisements	Linux Host	nstat
Icmp6InRedirects	Linux Host	nstat
Icmp6InMLDv2Reports	Linux Host	nstat
Icmp6OutDestUnreachs	Linux Host	nstat
Icmp6OutPktTooBigs	Linux Host	nstat
Icmp6OutTimeExcds	Linux Host	nstat
Icmp6OutParmProblems	Linux Host	nstat
Icmp6OutEchos	Linux Host	nstat
Icmp6OutEchoReplies	Linux Host	nstat
Icmp6OutGroupMembQueries	Linux Host	nstat
Icmp6OutGroupMembResponses	Linux Host	nstat
Icmp6OutGroupMembReductions	Linux Host	nstat
Icmp6OutRouterSolicits	Linux Host	nstat

Continued on next page

Table A.7 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
Icmp6OutRouterAdvertisements	Linux Host	nstat
Icmp6OutNeighborSolicits	Linux Host	nstat
Icmp6OutNeighborAdvertisements	Linux Host	nstat
Icmp6OutRedirects	Linux Host	nstat
Icmp6OutMLDv2Reports	Linux Host	nstat
Icmp6InType134	Linux Host	nstat
Icmp6InType143	Linux Host	nstat
Icmp6OutType133	Linux Host	nstat
Icmp6OutType135	Linux Host	nstat
Icmp6OutType143	Linux Host	nstat
Udp6InDatagrams	Linux Host	nstat
Udp6NoPorts	Linux Host	nstat
Udp6InErrors	Linux Host	nstat
Udp6OutDatagrams	Linux Host	nstat
Udp6RcvbufErrors	Linux Host	nstat
Udp6SndbufErrors	Linux Host	nstat
Udp6InCsumErrors	Linux Host	nstat
Udp6IgnoredMulti	Linux Host	nstat
UdpLite6InDatagrams	Linux Host	nstat
UdpLite6NoPorts	Linux Host	nstat
UdpLite6InErrors	Linux Host	nstat
UdpLite6OutDatagrams	Linux Host	nstat
UdpLite6RcvbufErrors	Linux Host	nstat
UdpLite6SndbufErrors	Linux Host	nstat
UdpLite6InCsumErrors	Linux Host	nstat

Continued on next page

Table A.7 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
TcpExtSyncookiesSent	Linux Host	nstat
TcpExtSyncookiesRecv	Linux Host	nstat
TcpExtSyncookiesFailed	Linux Host	nstat
TcpExtEmbryonicRsts	Linux Host	nstat
TcpExtPruneCalled	Linux Host	nstat
TcpExtRcvPruned	Linux Host	nstat
TcpExtOfoPruned	Linux Host	nstat
TcpExtOutOfWindowIcmps	Linux Host	nstat
TcpExtLockDroppedIcmps	Linux Host	nstat
TcpExtArpFilter	Linux Host	nstat
TcpExtTW	Linux Host	nstat
TcpExtTWRecycled	Linux Host	nstat
TcpExtTWKilled	Linux Host	nstat
TcpExtPAWSActive	Linux Host	nstat
TcpExtPAWSEstab	Linux Host	nstat
TcpExtDelayedACKs	Linux Host	nstat
TcpExtDelayedACKLocked	Linux Host	nstat
TcpExtDelayedACKLost	Linux Host	nstat
TcpExtListenOverflows	Linux Host	nstat
TcpExtListenDrops	Linux Host	nstat
TcpExtTCPHPHits	Linux Host	nstat
TcpExtTCPPureAcks	Linux Host	nstat
TcpExtTCPHPAcks	Linux Host	nstat
TcpExtTCPRenoRecovery	Linux Host	nstat
TcpExtTCPSackRecovery	Linux Host	nstat

Continued on next page

Table A.7 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
TcpExtTCPSACKReneging	Linux Host	nstat
TcpExtTCPSACKReorder	Linux Host	nstat
TcpExtTCPRenoReorder	Linux Host	nstat
TcpExtTCPTSReorder	Linux Host	nstat
TcpExtTCPFullUndo	Linux Host	nstat
TcpExtTCPPartialUndo	Linux Host	nstat
TcpExtTCPDSACKUndo	Linux Host	nstat
TcpExtTCPLossUndo	Linux Host	nstat
TcpExtTCPLostRetransmit	Linux Host	nstat
TcpExtTCPRenoFailures	Linux Host	nstat
TcpExtTCPSackFailures	Linux Host	nstat
TcpExtTCPLossFailures	Linux Host	nstat
TcpExtTCPFastRetrans	Linux Host	nstat
TcpExtTCPSlowStartRetrans	Linux Host	nstat
TcpExtTCPTimeouts	Linux Host	nstat
TcpExtTCPLossProbes	Linux Host	nstat
TcpExtTCPLossProbeRecovery	Linux Host	nstat
TcpExtTCPRenoRecoveryFail	Linux Host	nstat
TcpExtTCPSackRecoveryFail	Linux Host	nstat
TcpExtTCPRcvCollapsed	Linux Host	nstat
TcpExtTCPDSACKOldSent	Linux Host	nstat
TcpExtTCPDSACKOfoSent	Linux Host	nstat
TcpExtTCPDSACKRecv	Linux Host	nstat
TcpExtTCPDSACKOfoRecv	Linux Host	nstat
TcpExtTCPAbortOnData	Linux Host	nstat

Continued on next page

Table A.7 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
TcpExtTCPAbortOnClose	Linux Host	nstat
TcpExtTCPAbortOnMemory	Linux Host	nstat
TcpExtTCPAbortOnTimeout	Linux Host	nstat
TcpExtTCPAbortOnLinger	Linux Host	nstat
TcpExtTCPAbortFailed	Linux Host	nstat
TcpExtTCPMemoryPressures	Linux Host	nstat
TcpExtTCPMemoryPressuresChrono	Linux Host	nstat
TcpExtTCPSACKDiscard	Linux Host	nstat
TcpExtTCPDSACKIgnoredOld	Linux Host	nstat
TcpExtTCPDSACKIgnoredNoUndo	Linux Host	nstat
TcpExtTCPSpuriousRTOs	Linux Host	nstat
TcpExtTCPMD5NotFound	Linux Host	nstat
TcpExtTCPMD5Unexpected	Linux Host	nstat
TcpExtTCPMD5Failure	Linux Host	nstat
TcpExtTCPSackShifted	Linux Host	nstat
TcpExtTCPSackMerged	Linux Host	nstat
TcpExtTCPSackShiftFallback	Linux Host	nstat
TcpExtTCPBacklogDrop	Linux Host	nstat
TcpExtPFMemallocDrop	Linux Host	nstat
TcpExtTCPMinTTLDrop	Linux Host	nstat
TcpExtTCPDeferAcceptDrop	Linux Host	nstat
TcpExtIPReversePathFilter	Linux Host	nstat
TcpExtTCPTimeWaitOverflow	Linux Host	nstat
TcpExtTCPReqQFullDoCookies	Linux Host	nstat
TcpExtTCPReqQFullDrop	Linux Host	nstat

Continued on next page

Table A.7 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
TcpExtTCPRetransFail	Linux Host	nstat
TcpExtTCPRcvCoalesce	Linux Host	nstat
TcpExtTCPOFOQueue	Linux Host	nstat
TcpExtTCPOFODrop	Linux Host	nstat
TcpExtTCPOFOMerge	Linux Host	nstat
TcpExtTCPChallengeACK	Linux Host	nstat
TcpExtTCPSYNChallenge	Linux Host	nstat
TcpExtTCPFastOpenActive	Linux Host	nstat
TcpExtTCPFastOpenActiveFail	Linux Host	nstat
TcpExtTCPFastOpenPassive	Linux Host	nstat
TcpExtTCPFastOpenPassiveFail	Linux Host	nstat
TcpExtTCPFastOpenListenOverflow	Linux Host	nstat
TcpExtTCPFastOpenCookieReqd	Linux Host	nstat
TcpExtTCPFastOpenBlackhole	Linux Host	nstat
TcpExtTCPSpuriousRtxHostQueues	Linux Host	nstat
TcpExtBusyPollRxPackets	Linux Host	nstat
TcpExtTCPAutoCorking	Linux Host	nstat
TcpExtTCPFromZeroWindowAdv	Linux Host	nstat
TcpExtTCPToZeroWindowAdv	Linux Host	nstat
TcpExtTCPWantZeroWindowAdv	Linux Host	nstat
TcpExtTCPSynRetrans	Linux Host	nstat
TcpExtTCPOrigDataSent	Linux Host	nstat
TcpExtTCPHystartTrainDetect	Linux Host	nstat
TcpExtTCPHystartTrainCwnd	Linux Host	nstat
TcpExtTCPHystartDelayDetect	Linux Host	nstat

Continued on next page

Table A.7 – *Continued from previous page*

Data Element	Vantage Point	Collection Method
TcpExtTCPHystartDelayCwnd	Linux Host	nstat
TcpExtTCPACKSkippedSynRecv	Linux Host	nstat
TcpExtTCPACKSkippedPAWS	Linux Host	nstat
TcpExtTCPACKSkippedSeq	Linux Host	nstat
TcpExtTCPACKSkippedFinWait2	Linux Host	nstat
TcpExtTCPACKSkippedTimeWait	Linux Host	nstat
TcpExtTCPACKSkippedChallenge	Linux Host	nstat
TcpExtTCPWinProbe	Linux Host	nstat
TcpExtTCPKeepAlive	Linux Host	nstat
TcpExtTCPMTUPFail	Linux Host	nstat
TcpExtTCPMTUPSuccess	Linux Host	nstat
Netflow Records	Linux Host, Switch, etc.	Various

Table A.8: Layer 4 Statistics Description

Data Element	Stat Description
IcmpInMsgs	ICMP packets received
IcmpInErrors	ICMP packets with errors
IcmpInDestUnreachs	ICMP packets with no route to destination
IcmpInTimeExcds	TTL exceeded messages received
IcmpInParmProbs	Bad IP headers
IcmpInSrcQuenchs	Packet lost messages slow down
IcmpInRedirects	Error message to use a new route to same destination
IcmpInEchos	Echos of ICMP messages received

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	**Stat Description**
IcmpInEchoReps	ICMP echo replies
IcmpInTimestamps	ICMP timestamp request
IcmpInTimestampReps	Timestamp replies
IcmpInAddrMasks	Address mask request
IcmpInAddrMaskReps	Address mask replies
IcmpOutMsgs	ICMP packets sent
IcmpOutErrors	ICMP packets sent with errors
IcmpOutDestUnreachs	ICMP packets with no route to destination
IcmpOutTimeExcds	Packets sent that expire (TTL = 0)
IcmpOutParmProbs	Bad IP headers
IcmpOutSrcQuenchs	Packet lost messages slow down sent
IcmpOutRedirects	Error message to use a new route to same destination sent
IcmpOutEchos	Echos sent
IcmpOutEchoReps	Echo replies sent
IcmpOutTimestamps	Timestamp requests sent
IcmpOutTimestampReps	Timestamp replies sent
IcmpOutAddrMasks	Address mask requests sent
IcmpInMsgs	ICMP packet received
IcmpInErrors	ICMP packets received with errors
IcmpInCsumErrors	Packets with bad checksums
IcmpInDestUnreachs	ICMP packets with no route to destination
IcmpInTimeExcds	TTL exceeded messages received
IcmpInParmProbs	Bad IP headers
IcmpInSrcQuenchs	Packet lost messages slow down

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	Stat Description
IcmpInRedirects	Error message to use a new route to same destination
IcmpInEchos	Echos of ICMP messages received
IcmpInEchoReps	ICMP echo replies
IcmpInTimestamps	ICMP timestamp request
IcmpInTimestampReps	Timestamp replies
IcmpInAddrMasks	Address mask request
IcmpInAddrMaskReps	Address mask replies
IcmpOutMsgs	ICMP packets sent
IcmpOutErrors	ICMP packets sent with errors
IcmpOutDestUnreachs	ICMP packets with no route to destination
IcmpOutTimeExcds	Packets sent that expire (TTL = 0)
IcmpOutParmProbs	Bad IP headers
IcmpOutSrcQuenchs	Packet lost messages slow down sent
IcmpOutRedirects	Error message to use a new route to same destination sent
IcmpOutEchos	Echos sent
IcmpOutEchoReps	Echo replies sent
IcmpOutTimestamps	Timestamp requests sent
IcmpOutTimestampReps	Timestamp replies sent
IcmpOutAddrMasks	Address mask requests sent
IcmpOutAddrMaskReps	Address mask replies sent
IcmpMsgInType3	Destination unreachable
IcmpMsgInType11	Time exceeded
IcmpMsgOutType3	Destination unreachable

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	**Stat Description**
IcmpMsgOutType11	Time exceeded
TcpActiveOpens	The number of times that TCP connections have made a direct transition to the SYN-SENT state from the CLOSED state
TcpPassiveOpens	The number of times TCP connections have made a direct transition to the SYN-RCVD state from the LISTEN state
TcpAttemptFails	The number of times that TCP connections have made a direct transition to the CLOSED state from either the SYN-SENT state or the SYN-RCVD state, plus the number of times that TCP connections have made a direct transition to the LISTEN state from the SYN-RCVD state
TcpEstabResets	The number of times that TCP connections have made a direct transition to the CLOSED state from either the ESTABLISHED state or the CLOSE-WAIT state
TcpInSegs	TCP segments received
TcpOutSegs	TCP segments sent
TcpRetransSegs	TCP retransmitted segments
TcpInErrs	TCP segments received with errors
TcpOutRsts	TCP segments sent with reset flag
TcpInCsumErrors	TCP segments received with bad checksums
UdpInDatagrams	UDP datagrams received

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	**Stat Description**
UdpNoPorts	The total number of received UDP datagrams for which there was no application at the destination port
UdpInErrors	UDP datagrams received with errors
UdpOutDatagrams	UDP datagrams sent
UdpRcvbufErrors	UDP receive buffer errors
UdpSndbufErrors	UDP send buffer errors
UdpInCsumErrors	UDP datagrams received with bad checksum
UdpIgnoredMulti	UDP ignored multicasts
UdpLiteInDatagrams	UDP lite datagrams received
UdpLiteNoPorts	The total number of received UDP lite datagrams for which there was no application at the destination port
UdpLiteInErrors	UDP lite datagrams received with errors
UdpLiteOutDatagrams	UDP lite datagrams sent
UdpLiteRcvbufErrors	UDP lite receive buffer errors
UdpLiteSndbufErrors	UDP lite send buffer errors
UdpLiteInCsumErrors	UDP lite datagrams received with bad checksum
UdpLiteIgnoredMulti	UDP lite ignored multicasts
Icmp6InMsgs	ICMP packets received using IPv6
Icmp6InErrors	ICMP packets received with error using IPv6
Icmp6OutMsgs	ICMP packets sent using IPv6
Icmp6OutErrors	ICMP packets sent with errors using IPv6

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	Stat Description
Icmp6InCsumErrors	ICMP packets received with bad checksums using IPv6
Icmp6InDestUnreachs	ICMP destination unreachable messages received using IPv6
Icmp6InPktTooBigs	ICMP packet too big messages received using IPv6
Icmp6InTimeExcds	ICMP packet timeout messages received using IPv6
Icmp6InParmProblems	ICMP packets with bad IP headers using IPv6
Icmp6InEchos	ICMP echo received using IPv6
Icmp6InEchoReplies	ICMP echo replies received using IPv6
Icmp6InGroupMembQueries	ICMP group member queries received using IPv6
Icmp6InGroupMembResponses	ICMP group member query responses received using IPv6
Icmp6InGroupMembReductions	ICMP group reduction messages received using IPv6
Icmp6InRouterSolicits	ICMP router solicitation messages received using IPv6
Icmp6InRouterAdvertisements	ICMP router advertisement messages received using IPv6
Icmp6InNeighborSolicits	ICMP messages received sent to locate routers using IPv6

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	Stat Description
Icmp6InNeighborAdvertisements	ICMP messages received from routers to advertise existence using IPv6
Icmp6InRedirects	ICMP messages received sent to reroute more efficiently
Icmp6InMLDv2Reports	Multicast Listener Discovery reports
Icmp6OutDestUnreachs	ICMP destination unreachable messages sent using IPv6
Icmp6OutPktTooBigs	ICMP packet too big messages sent using IPv6
Icmp6OutTimeExcds	ICMP packet timeout messages sent using IPv6
Icmp6OutParmProblems	ICMP packets with bad IP headers using IPv6
Icmp6OutEchos	ICMP echo sent using IPv6
Icmp6OutEchoReplies	ICMP echo replies sent using IPv6
Icmp6OutGroupMembQueries	ICMP group member queries sent using IPv6
Icmp6OutGroupMembResponses	ICMP group member query responses sent using IPv6
Icmp6OutGroupMembReductions	ICMP group reduction messages sent using IPv6
Icmp6OutRouterSolicits	ICMP router solicitation messages sent using IPv6
Icmp6OutRouterAdvertisements	ICMP router advertisement messages sent using IPv6

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	Stat Description
Icmp6OutNeighborSolicits	ICMP messages sent to locate routers using IPv6
Icmp6OutNeighborAdvertisements	ICMP messages sent to neighbors to advertise existence
Icmp6OutRedirects	ICMP messages sent to reroute more efficiently
Icmp6OutMLDv2Reports	Multicast Listener Discovery reports sent
Icmp6InType134	Router advertisements messages received
Icmp6InType143	Multicast listener reports received
Icmp6OutType133	Router solicitation messages sent
Icmp6OutType135	Neighbor solicitation messages sent
Icmp6OutType143	Multicast listener reports sent
Udp6InDatagrams	UDP datagrams received using IPv6
Udp6NoPorts	The total number of received UDP datagrams for which there was no application at the destination port using IPv6
Udp6InErrors	UDP datagrams received with errors using IPv6
Udp6OutDatagrams	UDP datagrams sent using IPv6
Udp6RcvbufErrors	UDP receive buffer errors using IPv6
Udp6SndbufErrors	UDP send buffer errors using IPv6
Udp6InCsumErrors	UDP datagrams received with bad checksum using IPv6
Udp6IgnoredMulti	UDP ignored multicasts using IPv6
UdpLite6InDatagrams	UDP lite datagrams received using IPv6

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	Stat Description
UdpLite6NoPorts	The total number of received UDP lite datagrams for which there was no application at the destination port using IPv6
UdpLite6InErrors	UDP lite datagrams received with errors using IPv6
UdpLite6OutDatagrams	UDP lite datagrams sent using IPv6
UdpLite6RcvbufErrors	UDP lite receive buffer errors using IPv6
UdpLite6SndbufErrors	UDP lite send buffer errors using IPv6
UdpLite6InCsumErrors	UDP lite datagrams received with bad checksum using IPv6
TcpExtSyncookiesSent	TCP SYN cookies sent
TcpExtSyncookiesRecv	TCP SYN cookies received
TcpExtSyncookiesFailed	Bad TCP SYN cookies
TcpExtEmbryonicRsts	TCP connection that has not finished handshake process reset
TcpExtPruneCalled	TCP pruning process has been called
TcpExtRcvPruned	TCP packets received that have been pruned
TcpExtOfoPruned	Out of order TCP packets that have been pruned
TcpExtOutOfWindowIcmps	ICMP request that are out of appropriate window
TcpExtLockDroppedIcmps	ICMP messages that have been locked and dropped
TcpExtArpFilter	ARP messages that have been filtered
TcpExtTW	Time waits initialized

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	**Stat Description**
TcpExtTWRecycled	Time waits recycled
TcpExtTWKilled	Time waits stopped
TcpExtPAWSActive	TCP Protection Against Wrapping Sequence active connections
TcpExtPAWSEstab	TCP Protection Against Wrapping Sequence established connections
TcpExtDelayedACKs	Delayed acknowledgments received
TcpExtDelayedACKLocked	Delayed acknowledgments locked
TcpExtDelayedACKLost	Delayed acknowledgments lost
TcpExtListenOverflows	Overflow of listening sockets
TcpExtListenDrops	Dropped listening sockets
TcpExtTCPHPHits	TCP HP Hits
TcpExtTCPPureAcks	Acknowledgments with no data
TcpExtTCPHPAcks	HP Acknowledgments
TcpExtTCPRenoRecovery	TCP Reno recovery invoked
TcpExtTCPSackRecovery	Selective acknowledgment recovery
TcpExtTCPSACKReneging	Selective acknowledgment that is later discarded
TcpExtTCPSACKReorder	Selective acknowledgment segment reordering
TcpExtTCPRenoReorder	TCP Reno segments reordered
TcpExtTCPTSReorder	TCP timestamp reordering
TcpExtTCPFullUndo	TCP complete undo
TcpExtTCPPartialUndo	TCP partial undo
TcpExtTCPDSACKUndo	TCP selective acknowledgment undo

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	Stat Description
TcpExtTCPLossUndo	TCP loss undo
TcpExtTCPLostRetransmit	TCP lost retransmissions
TcpExtTCPRenoFailures	TCP Reno segment failures
TcpExtTCPSackFailures	TCP selective acknowledgment failures
TcpExtTCPLossFailures	TCP loss failures
TcpExtTCPFastRetrans	TCP fast retransmissions
TcpExtTCPSlowStartRetrans	TCP retransmissions sent during slow start
TcpExtTCPTimeouts	TCP timeouts
TcpExtTCPLossProbes	TCP loss probes
TcpExtTCPLossProbeRecovery	TCP loss probes recovered
TcpExtTCPRenoRecoveryFail	TCP Reno recovery failed
TcpExtTCPSackRecoveryFail	TCP selective acknowledgment recovery failures
TcpExtTCPRcvCollapsed	TCP packets that have been collapsed
TcpExtTCPDSACKOldSent	Old duplicate selective acknowledgments
TcpExtTCPDSACKOfoSent	Out of order duplicate selective acknowledgments
TcpExtTCPDSACKRecv	Duplicate selective acknowledgments received
TcpExtTCPDSACKOfoRecv	Out of order duplicate selective acknowledgments received
TcpExtTCPAbortOnData	TCP connection abort due to no/bad data
TcpExtTCPAbortOnClose	TCP connection abort due to closed socket
TcpExtTCPAbortOnMemory	TCP connection abort due to no memory
TcpExtTCPAbortOnTimeout	TCP connection abort due to timeout

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	Stat Description
TcpExtTCPAbortOnLinger	TCP connection abort due to lingering connection
TcpExtTCPAbortFailed	TCP connection aborts failed
TcpExtTCPMemoryPressures	TCP memory pressure flags raised
TcpExtTCPMemoryPressuresChrono	TCP memory pressure flags chrono
TcpExtTCPSACKDiscard	TCP selective acknowledgment discards
TcpExtTCPDSACKIgnoredOld	TCP duplicate selective acknowledgments ignored
TcpExtTCPDSACKIgnoredNoUndo	TCP duplicate selective acknowledgments ignored that haven't been undone
TcpExtTCPSpuriousRTOs	TCP retransmission timeouts
TcpExtTCPMD5NotFound	MD5 Checksum not found
TcpExtTCPMD5Unexpected	Unexpected MD5 checksum
TcpExtTCPMD5Failure	MD5 checksum failure
TcpExtTCPSackShifted	Selective acknowledgments shifted
TcpExtTCPSackMerged	Selective acknowledgments merged
TcpExtTCPSackShiftFallback	Selective acknowledgment shift fallbacks
TcpExtTCPBacklogDrop	Dropped connections due to TCP backlog
TcpExtPFMemallocDrop	Dropped connections due to memory allocation problems
TcpExtTCPMinTTLDrop	Minimum time to live connection drops
TcpExtTCPDeferAcceptDrop	Connections dropped after deferred acceptation
TcpExtIPReversePathFilter	Packets dropped when they exit a different interface than they came in on

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	Stat Description
TcpExtTCPTimeWaitOverflow	Time wait overflow
TcpExtTCPReqQFullDoCookies	TCP requested queue cookies
TcpExtTCPReqQFullDrop	TCP requested full queue drop
TcpExtTCPRetransFail	TCP retransmissions failed
TcpExtTCPRcvCoalesce	TCP segments coalesced
TcpExtTCPOFOQueue	TCP out of order queue count
TcpExtTCPOFODrop	TCP out of order drops
TcpExtTCPOFOMerge	TCP out of order merges
TcpExtTCPChallengeACK	TCP challenge acknowledgments
TcpExtTCPSYNChallenge	TCP SYN challenges
TcpExtTCPFastOpenActive	Number of fast open connections that are active
TcpExtTCPFastOpenActiveFail	Fast open connections that have failed
TcpExtTCPFastOpenPassive	Fast open connections that are connected but passive
TcpExtTCPFastOpenPassiveFail	Passive fast open connections that have failed
TcpExtTCPFastOpenListenOverflow	Fast open connections dropped from too many connections trying to be made
TcpExtTCPFastOpenCookieReqd	Fast open connection cookies requested
TcpExtTCPFastOpenBlackhole	Perpetual failure of TCP fast open recovery
TcpExtTCPSpuriousRtxHostQueues	Spurious host queues
TcpExtBusyPollRxPackets	Packets busy polling
TcpExtTCPAutoCorking	Sockets corked automatically
TcpExtTCPFromZeroWindowAdv	No space left in receive buffer to send

Continued on next page

Table A.8 – *Continued from previous page*

Data Element	**Stat Description**
TcpExtTCPToZeroWindowAdv	No space left in receive buffer to receive
TcpExtTCPWantZeroWindowAdv	Message to fill up receive buffer
TcpExtTCPSynRetrans	TCP SYN retransmissions
TcpExtTCPOrigDataSent	Bytes of original TCP data sent
TcpExtTCPHystartTrainDetect	Hystart congestion detection
TcpExtTCPHystartTrainCwnd	Hystart congestion window size update
TcpExtTCPHystartDelayDetect	Delayed Hystart congestion detection
TcpExtTCPHystartDelayCwnd	Delayed Hystart congestion window size update
TcpExtTCPACKSkippedSynRecv	Skipped SYNs received
TcpExtTCPACKSkippedPAWS	Skipped protection against wrapping sequence acknowledgments skipped
TcpExtTCPACKSkippedSeq	Skipped sequence acknowledgments
TcpExtTCPACKSkippedFinWait2	Skipped FIN acknowledgments during wait
TcpExtTCPACKSkippedTimeWait	Skipped time wait acknowledgments
TcpExtTCPACKSkippedChallenge	Skipped challenge acknowledgments
TcpExtTCPWinProbe	Window probes sent
TcpExtTCPKeepAlive	Keep connection alive messages sent
TcpExtTCPMTUPFail	TCP maximum transfer unit packet fails
TcpExtTCPMTUPSuccess	TCP maximum transfer unit packet successes
Netflow Records	IP statistics of flows

Curriculum Vitae

Christopher Mendoza began his Electrical Engineering undergraduate degree at the University of Texas at El Paso (UTEP) in 2012 and finished in 2017 and promptly followed to attend graduate school also at UTEP. His undergraduate senior project, an innovative suturing simulator, resulted in a patent application. He currently works as a graduate research assistant in the UTEP communication networks lab (NetLab). His main research interests are in communication networks, data analysis, machine learning and fault detection.

Email: camendoza7@miners.utep.edu