Mechanisms and Protocols for Reliable Communication Networks

Désirée Rentz, Henning Stubbe*, Kilian Holzinger*
*Chair of Network Architectures and Services, Department of Informatics
Technical University of Munich, Germany
Email: desiree.rentz@tum.de, stubbe@net.in.tum.de, holzinger@net.in.tum.de

Abstract—**When creating communication networks, technical problems occur that can disrupt or even break the communication. To overcome the technical challenge of making such communication reliable, various mechanisms have been invented. After mentioning some common failures that negatively affect the functionality of a network, this paper summarizes a selection of mechanisms that have been created to improve the reliability of communication networks, and explains some metrics that can be used to characterize and compare these mechanisms.**

Index Terms—**reliability, time sensitive networking, interference, congestion**

1. Introduction

There are many different examples for communication networks, such as ACARS, which is used in aviation, or DVB-S2, a standard for satellite television broadcast. This paper discusses the reliability mechanisms used for the Internet. However, their application is not restricted to that; for example the error correcting code briefly described in Section 3.1 is also used in DVB-S2. [1], [2]

1.1. Definition of Reliability

Since there is no generally applicable definition of reliability, we will first examine some aspects contributing to it.

From a more abstract perspective, Zhang et al. divide the analysis of network reliability into three layers: connectivity, performance and application reliability. The first refers to network topology and physical connectivity, the second is described as the "probability that performance indicators remain their values within expected ranges under a certain traffic flow" [3]. In their work on network reliability testing, Li et al. describe network reliability as the ability to ensure the functionality of the network. They summarize this as "transmitting data timely, completely and correctly", which are measurable quantities [4]. In another paper dealing with wireless network routing, Biswas et al. describe reliability as "a mission-specific metric evaluating the probability that a packet gets delivered with a given deadline". They consider it "to be a combination of availability and dependability" [5]. Similarly, Shi et al. consider the "reliability of a network defined as the probability of successful communication" [6].

Based on these statements, we can conclude that in order for a communication network to be considered reliable, it must offer high availability (i.e. data transmission to the destination must be possible at any given point in time), and correct, complete and timely transmission must be ensured. The degree to which the latter three requirements must be fulfilled depends on the application.

To avoid ambiguity, it should be noted that this does not include performance increases beyond the absolute minimum that is defined by the application. The remainder of this paper also assumes that there are no intentional attacks on the reliability of the network.

1.2. Types of Failures

The first step of improving the reliability is to analyze the typical impediments during data transmission. Shi et al. summarize that "the failures of computer networks usually consist of two modes: connective failures and congestion failures" [6].

Due to the limitations of the network participants, too much traffic can lead to congestion failures. These can manifest in the form of lost or delayed packets or failing to establish new connections in connection-oriented protocols.

Connectivity failures are failures in which the transmission of data between two nodes is compromised. The data arriving at the receiving end may contain bit errors of which the receiver is unaware. This can range from a single flipped bit to completely corrupted data. When the quality of a link has degraded to the point where data can no longer be transmitted, this is referred to as a link failure.

2. Background

In preparation for the next section, this section will first provide a general overview of the relevant protocols and provide some background knowledge.

The inner working of the Internet is realized through a variety of protocols. The protocols used in Internet communication are usually categorized according to the OSI model, in which each protocol of an upper layer builds on the layers below it, with the lowest layer being the physical communication. The layer above does not necessarily need to know about the implementation of the layer below.

Figure 1 contains the protocols that are used as examples in the following. The model provides a rough overview, but protocols cannot always be clearly assigned to a layer. Some protocol specifications may span multiple layers, as it is the case with Ethernet. Other protocols work within a layer, together with another protocol. For

doi: 10.2313/NET-2023-06-1_11

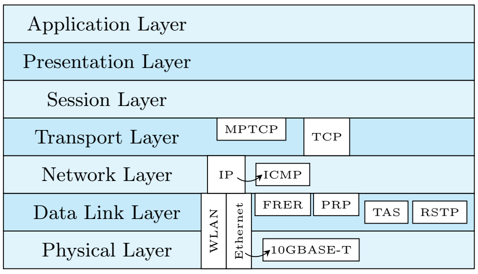


Figure 1: OSI model with exemplified protocols

example, MPTCP can optionally be stacked onto TCP, and still operate within the transport layer.

Ethernet (IEEE 802.3). Ethernet is a protocol standardized by the IEEE 802.3 working group. Its specification covers both the physical and data link layer and is used for Local Area Networks and Metropolean Area Networks (LAN/MAN). The physical layer uses wired connections, and offers multiple implementations for copper and fiber wiring using both full-duplex and half-duplex configurations. For example, 10GBASE-T Ethernet is based on a full-duplex copper medium, and supports a data rate of 10 Gbit/s. [7, Clause 1 and 55]

WLAN (IEEE 802.11). The IEEE 802.11 working group creates standards to implement WLANs (Wireless Local Area Networks). The WLAN specification covers both the physical and data link layer. As with Ethernet, several options are defined for the physical layer, using different frequency bands. Wireless communication is half-duplex and generally more susceptible to bit errors compared to wired media. [8, Clause 4 and 8]

Time Sensitive Networking (IEEE 802.1 TSN). One part of the IEEE 802.1 working group is the Time Sensitive Networking (TSN) task group. It develops a collection of standards aimed at "providing deterministic connectivity through IEEE 802 networks." Two standards we will look briefly at are IEEE 802.1CB (Frame Replication and Elimination for Reliability, FRER) and IEEE 802.1Qbv (Time Aware Shaper, TAS). [9]

Transmission Control Protocol (TCP). The Transmission Control Protocol (TCP) was developed together with the Internet Protocol (IP). It enables stream-oriented data transmission using connections. Since IP explicitly does not implement reliability mechanisms, TCP makes very few assumptions about the reliability of the underlying layers. [10], [11]

3. Mechanisms to Improve Reliability

In the following, we will look at certain types of failures and the corresponding mechanisms.

3.1. Dealing with Interference

Electronic interference can be caused by external sources, but also by the electronic equipment itself. It usually leads to bit errors. A special type of interference is crosstalk, where two parallel data lines influence each other due to electromagnetism.

Physical Wiring. A very common strategy to minimize the crosstalk between two parallel pairs of wires within a cable is to twist the wires inside around each other. Additionally, the interference from external sources can be reduced by shielding the cable, using a woven copper shield, or wrapping the cable in foil. These cables are commonly referred to as Unshielded Twisted Pair (UTP) and Shielded Twisted Pair (STP) cable depending on whether shielding was used or not. In buildings, the physical placement of cables and wireless stations should also be considered, especially when sources of high interference exist. [12]

Signal Processing. If the interference is known to the transmitter, there are methods for the transmitter to precode a signal before sending. One type of precoding is known as Dirty Paper Coding (DPC). The fundamental idea is to modify the sent signal in such way that if the interference is added, the receiver will be unaware of the interference. One such DPC is the Tomlinson-Harashima precoder, which is used in 10GBASE-T Ethernet to cancel out near-end crosstalk, a type of crosstalk detected on the same side where the signal was sent. [12], [13], [7, Clause 55]

Error Correcting Codes. By encoding the digital data with an error correcting code before transmission, some bit error patterns can be corrected by the receiver. One example of an error correcting code is low density parity control (LDPC), which used in 10GBASE-T Ethernet. It calculates parity bits using a sparse matrix, where each data bit is included in at least two parity bits. [7, Clause 55], [14]

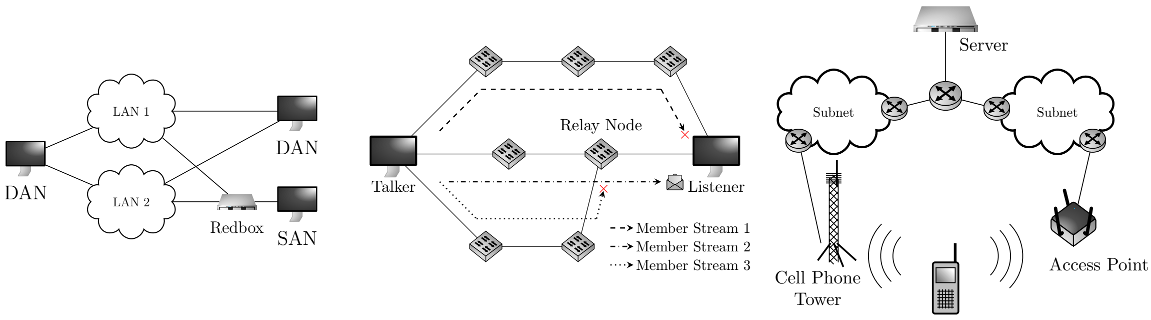
Error Detection Codes. In contrast to error correcting codes, error detection codes aim to detect as many error patterns as possible, but without being able to reconstruct the original data. One such code is the Cyclic Redundancy Check (CRC), which is based on polynomial division. At the data link layer, a 32-bit wide CRC code is used in the Frame Check Sequence field of the Ethernet and the WLAN header. [7, Clause 3], [8, Clause 9]

Additional Checksums. Several network and transport layer protocols implement an additional checksum value to verify the correctness of the received data. TCP, for example, calculates a checksum from a 16-bit wide sum of the data using one's complements. Other examples of third and fourth layer protocols that use checksums include UDP, IPv4 and ICMP. The usefulness of additional checksums has been confirmed in literature. [10], [11], [15]–[17]

3.2. Dealing with Collisions

If two nodes send data at the same time in half-duplex configurations, the signals overlap and become unusable. This is commonly referred to as collision.

CSMA/CD in Half-Duplex Ethernet. Ethernet implements a mechanism called CSMA/CD that aims to avoid collisions in half-duplex setups. The algorithm can be thought of as multiple people (Multiple Access) talking



(a) Parallel Redundancy Protocol (b) Frame Elimination and Replication (c) Multipath TCP

Figure 2: Redundant path usage with different protocols

at a table: Whenever a person has something to say, they check whether anyone else is speaking, before they talk (Carrier Sense). If two people start talking at the same time, they stop, wait for a randomly chosen time interval, and then try again (Collision Detection). [18]

However, today's networking devices most commonly use the full-duplex mode. Full-duplex does not require the use of CSMA/CD, since collisions cannot occur.

CSMA/CA in WLAN. While collisions can be fully avoided in Ethernet using full-duplex connections, a wireless network is always half-duplex. Unlike wired connections, the medium cannot be listened during transmission. Therefore, CSMA/CD cannot be directly applied to WLAN.

As with CSMA/CD, the medium is checked before transmission is attempted. If the channel is free, a distributed coordination function determines a time interval that the endpoint waits before attempting to send, to give other members in the channel a chance (Collision Avoidance). The waiting time consists of a randomized time interval and a constant interframe space.

The proper detection of whether the channel is clear may not be possible if a transmitting station is out of range of the station initiating the transmission. Both endpoints may be within range of the WLAN base station but not within range of each other. This is commonly referred to as the hidden station problem. To avoid collisions caused by this, an additional method can be implemented in which an endpoint in the network sends a request-to-send (RTS) frame to which the base station will reply with a clear-to-send (CTS) frame when permission is granted to transmit data. [19]

3.3. Dealing with Link Failures

When a link in the network becomes unusable, it is called a link failure. The consequence of this is a changed topology, which in turn changes the routing decisions that the nodes on the path must make. Usually, nodes communicate to each other about link failures using routing protocols.

Rapid Spanning Tree Protocol. The Rapid Spanning Tree Protocol (RSTP) is defined in IEEE 802.1w. Its purpose is to create a spanning tree from a network, i.e. to reduce the network topology to a tree so that any two nodes are connected by a single path only. In the event of a link failure, it adapts to the changed topology. Compared to its predecessor, the Spanning Tree Protocol (STP), RSTP is able to react much faster to topology changes. [20]

Parallel Redundancy Protocol. The Parallel Redundancy Protocol (PRP) is defined as part of an IEC norm. It is intended to guarantee no latency in the event of a link failure by sending each frame over multiple paths. PRP builds on Ethernet and identifies related frames using an additional header attached to the PDU.

PRP imposes requirements on the network topology, as shown in Figure 2a. Each node in the network should be connected to two separate LANs. These Doubly Attached Nodes (DANs) require hardware support. If a node does not support this, a Single Attached Node (SAN) can be placed behind a redundancy box (Redbox) to achieve the dual connectivity. The Doubly Attached Nodes and Redboxes are capable of sending and receiving duplicate frames, and eliminating them if necessary. In the event of a single network link failure, the frame arrives at its destination without any latency, as it would be the case with RSTP, because of these hardware requirements. [21]

Frame Replication and Elimination. Frame Replication and Elimination (FRER) is part of the IEEE 802.1 TSN specifications. Similar to PRP, it is designed to send frames over multiple paths (replication) and to provide a mechanism to eliminate duplicates at the receiving end (elimination). Again, the goal is to avoid any latency in the event of a link failure. FRER does not impose any topology requirements, but redundant hardware paths are required for it to have any effect.

FRER calls a series of frames from a talker to one or more listeners a component stream. A component stream is split into multiple member streams, with each member stream sent over a different path by duplicating the frames. When a relay or listener node receives duplicate frames, it will eliminate one of them, as shown in Figure 2b. FRER identifies the member streams by a so-called redundancy tag or by using a PRP header. In this regard, PRP and FRER are partially compatible with each other. [22], [23]

Multipath TCP. Consider the use case of a person walking from one building to another with a cell phone. As shown in Figure 2c, a cell phone has two interfaces through which it can connect to the Internet, either through a WLAN access point or through a cell phone tower. Depending on the location of the person holding the device, one signal may be stronger than the other.

With connection-oriented protocols, it is not as easy to switch to a different interface because TCP is not designed to handle changing IP addresses. This is where Multipath TCP (MPTCP), an extension of TCP, comes into play: It enables the use of redundant paths by using two or more interfaces of a device for a single connection, which may have different IP addresses. MPTCP keeps track of IP addresses using subflows. It introduces additional header fields that are prepended to the transport layer PDU. [24]

3.4. Retransmissions

Despite the possibility to repair or reroute data, sometimes data is simply lost irretrievably. In this case, the data should be retransmitted automatically. The methods for doing this are usually called Automatic Repeat Requests (ARQ) and use acknowledgement signals to confirm a successful transmission. For efficiency reasons, several data packets are usually sent at once. Sliding window algorithms such as Go-Back-N or Selective Repeat are used to keep track of unconfirmed data. They use a send window and a receive window to regulate the amount of data that is sent “at once”.

Retransmissions in WLAN. After a transmission with CSMA/CA, the transmitter alone cannot determine whether a collision has occurred or not. After sending the data, the sender waits for an ACK control frame from the access point. If the ACK control frame is not received, the sender retransmits the frame. A sender tries a configurable number of times before determining that communication has failed. For example, the implementation described in reference [25] does four or seven attempts, depending on the size of the PDU. This mechanism aims to create conditions similar to wired connections for the upper layers, since bit error are much more likely in the wireless medium. [8, Clause 4]

Retransmissions in TCP. With TCP, each byte sent between two nodes is acknowledged by the receiver using sequence numbers. The initial sequence numbers are exchanged during the three-way handshake. By observing the absence of acknowledgement frames, the sender can retransmit the data until the transmission was successful. The completeness of the transmission can be ensured by closing the connection, signaling the communication partner that the transmission is done. [11]

3.5. Dealing with Congestion

Congestion occurs when nodes on the network reach their limited capabilities, usually processing speed or buffer size. For example, a server in a data center may process data much faster than a mobile device, or a router in the network may become overwhelmed if it has to buffer too many packets at once, e.g. if data needs to be forwarded from a faster link to a slower link.

TCP Flow Control. The TCP flow control mechanism avoids congestion at the receiver. The receiver can announce the number of bytes it is willing to receive by using the receive window field in the TCP header. The sender then configures its sliding window procedure so that it does not send more than this number of bytes at once.

TCP Congestion Control. To avoid congestion in the network, TCP slowly increases the send window until it reaches the capacity that the network can currently handle. To test the capacity of the path, the transmit window is initially increased exponentially ("slow start"). After reaching the limit, which is usually determined by receiving multiple ACK signals acknowledging the same bytes, it halves the window and enters the "congestion avoidance" phase, where it approaches the capacity linearly.

Time Aware Shaping. TCP congestion control can prevent data loss due to congestion, but it cannot guarantee a maximum transmission time. Time Aware Shaping (TAS) addresses this problem by defining a different approach to media access at the data link layer by dividing the time on the channel into cycles, which in turn are divided into time slices. It introduces priorities for the frames and reserves one time slice for each priority within each cycle. In this way, if each hop to a destination keeps the link free for prioritized traffic, a maximum latency can be enforced as long as the prioritized traffic does not reach the limit of its own time slice. [26]

3.6. False Friends

It should be noted that some protocols implement mechanisms that at first glance are related to failures, but do not actively improve reliability. One such example is ICMP, which is part of IP and transmits error messages when an IP packet could not be delivered. IP itself is not a reliable protocol. The documentation for ICMP states, “The purpose of these control messages is to provide feedback about problems in the communications environment, not to make IP reliable.” [15]

4. Analysis

Table 1 gives an overview of which mechanism addresses which failure, and on which layer the corresponding protocol operates. From this we can now draw a few conclusions.

4.1. Evaluation

First of all, reliability problems are mostly due to hardware limitations. While some of these problems can be addressed directly, e.g. by improving the signal or increasing buffer sizes of the nodes, there is only so much that can be done on the physical layer. The advantage of the Internet’s multi-layered design is that even if these hardware-related errors cannot be fixed, the layers above them (primarily the data link and transport layer) can mitigate them. For this purpose, the protocols at the higher layers can implement one or more reliability mechanisms,

Mechanism	Failure	Layer	Technique	Improved Reliability Aspect
Physical Wiring	Interference	1	Prevention	Correctness
Dirty Paper Coding	Interference	1	Prevention	Correctness
Error Correcting Codes	Interference	1	Recovery	Correctness
Error Detection Code (CRC-32)	Interference	2	Detection	Correctness
Additional Checksum (TCP)	Interference	4	Detection	Correctness
CSMA/CD	Collision	2	Prevention, Detection	Correctness, Availability
CSMA/CA	Collision	2	Prevention	Correctness, Availability
Rapid Spanning Tree Protocol	Link Failure	2	Recovery	Availability
Parallel Redundancy Protocol	Link Failure	2.5	Recovery	Availability, Timeliness
Frame Replication and Elimination	Link Failure	2.5	Recovery	Availability, Timeliness
Multipath TCP	Link Failure	4.5	Recovery	Availability
TCP Retransmissions	Data Loss	4	Detection, Recovery	Completeness
WLAN Retransmissions	Data Loss	2	Detection, Recovery	Completeness
TCP Congestion Control	Congestion	4	Prevention, Detection	Availability
TCP Flow Control	Congestion	4	Prevention	Availability
Time Aware Shaping	Congestion	2	Prevention	Availability, Timeliness

TABLE 1: Summary of mechanisms discussed in this paper

as the examples of Ethernet and TCP show. There is not always a particular layer at which failures are addressed.

A mechanism usually targets a specific type of failure. There are several techniques how a failure can be addressed:

1) Prevention. For example, CSMA/CA attempts to prevent collisions.
2) Detection. Error detecting codes, checksums and acknowledgments can be used to detect if a failure occurred.
3) Recovery. These mechanisms attempt to cope with or correct a failure.

The underlying strategy to address a failure depends on the failure itself. Bit errors or link failures are addressed by adding redundancy in one form or another, half-duplex setups attempt to implement time division multiplexing, and congestion is avoided by limiting the amount of traffic in the network.

There is a relationship between the type of failure and the aspect of reliability that is improved. Previously, we defined reliability as complete, correct and timely data transmission, combined with high availability. Interference, link failures and congestion affect the availability of the network and the correctness of the data, so the mechanisms addressing them improve exactly these aspects of reliability. Retransmissions are special in that they target the symptom (= data loss) rather than a specific failure. Regardless of why the data was lost, these mechanisms attempt to recover the data. They are also the only mechanisms capable of improving or ensuring the completeness aspect of reliability.

It can be observed that the timeliness of data transmission requires dedicated mechanisms, since time guarantees are not provided by the common Ethernet/IP/TCP protocol stack. For many applications, a best-effort delivery is good enough, but if some applications have mission-specific Quality of Service (QoS) requirements, additional measures are needed.

4.2. Other Metrics

There are other metrics that can sometimes be used to further describe and compare reliability mechanisms.

Popularity. A very commonly used protocol stack is Ethernet/IP/TCP. This is reasonable, because every type of failure is addressed by at least one protocol. However, since they only provide best-effort data transmission, protocols like FRER and TAS that focus more on QoS have been developed. Also, high popularity does not imply high quality. A memo published by the IEC suggests, that there are CRC-32 polynoms with better properties than the one used in IEEE 802 networks. [27]

Overhead. For most protocols, only a few bytes of additional header fields or some acknowledgment signals are required, but apart from the implementation effort, this does not have too much impact. One influential overhead is hardware cost. While MPTCP can use existing interfaces, PRP requires an investment into dedicated hardware. A paper comparing FRER and PRP suggests that FRER is less expensive to implement. Some mechanisms may only be worth implementing for critical applications, but this is a tradeoff that must be decided for each application. [28]

Flexibility. Hardware setups are not only expensive but also inflexible, e.g. if full physical redundancy is required as with PRP, any host in the network must be connected twice, and at least twice as many network nodes and links have to be maintained. Higher-layer protocols may be more flexible, since they put less requirements on lower layers. They provide general purpose implementations. This may be the reason why TCP has remained incredibly popular since its introduction in 1981.

5. Conclusion

We have defined reliability in the context of communication networks and summarized what types of failures exist. Sixteen different mechanisms that improve the reliability have been described and categorized by the failures that they address and the layer in which they are implemented. Failures affect certain reliability aspects, such as the availability or correctness. Consequently, the mechanisms that address a specific failure improve exactly those aspects of reliability. Reliability mechanisms can prevent, detect, or recover from a failure. Three basic concepts for improving reliability are adding redundancy, providing a multiplexed setup, and not overloading the network capacity.

References

[1] S. Sun, "ACARS Data Identification and Application in Aircraft Maintenance," in *2009 First International Workshop on Database Technology and Applications*. IEEE, 2009, pp. 255–258.

[2] "Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2)," p. 22, 2009.

[3] H. Zhang, N. Huang, and H. Liu, "Network performance reliability evaluation based on network reduction," in *2014 Reliability and Maintainability Symposium*, 2014, pp. 1–6.

[4] R. Li, N. Huang, S. Li, R. Kang, and S. Chang, "Reliability Testing Technology for Computer Network Applications," in *2009 8th International Conference on Reliability, Maintainability and Safety*, 2009, pp. 1169–1172.

[5] T. Biswas, K. Lesser, R. Dutta, and M. Oishi, "Examining Reliability of Wireless Multihop Network Routing with Linear Systems," ser. HotSoS '14. New York, NY, USA: Association for Computing Machinery, 2014. [Online]. Available: https://doi.org/10.1145/2600176.2600195

[6] J. Shi, S. Wang, and K. Wang, "Congestion-Based Reliability Analysis for Computer Metworks," in *2009 8th International Conference on Reliability, Maintainability and Safety*, 2009, pp. 1149–1154.

[7] "IEEE Standard for Ethernet," *IEEE Std 802.3-2022 (Revision of IEEE Std 802.3-2018)*, pp. 1–7025, 2022.

[8] "IEEE Standard for Information Technology–Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks–Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-2020 (Revision of IEEE Std 802.11-2016)*, pp. 1–4379, 2021.

[9] "Welcome to the IEEE 802.1 Working Group," https://1.ieee802.org/, n.d., [Online; accessed 30-November-2022].

[10] "Internet Potocol," Internet Requests for Comments, RFC Editor, RFC 791, 1981. [Online]. Available: https://www.rfc-editor.org/rfc/rfc791.txt

[11] "Transmission Control Potocol," Internet Requests for Comments, RFC Editor, RFC 793, 1981. [Online]. Available: https://www.rfc-editor.org/rfc/rfc793.txt

[12] A. Oliviero and B. Woodward, *Cabling: The Complete Guide to Copper and Fiber-Optic Networking*, 4th ed. John Wiley & Sons, 2009.

[13] U. Erez and S. ten Brink, "A Close-to-Capacity Dirty Paper Coding Scheme," *IEEE Transactions on Information Theory*, vol. 51, no. 10, pp. 3417–3432, 2005.

[14] T. Strutz, "Low-Density Parity-Check Codes - An Introduction," 2016, [Online; accessed 7-December-2022].

[15] "Internet Control Message Protocol," Internet Requests for Comments, RFC Editor, RFC 792, 1981. [Online]. Available: https://www.rfc-editor.org/rfc/rfc792.txt

[16] "User Datagram Potocol," Internet Requests for Comments, RFC Editor, RFC 768, 1981. [Online]. Available: https://www.rfc-editor.org/rfc/rfc768.txt

[17] J. Stone and C. Partridge, "When the CRC and TCP Checksum Disagree," vol. 30, no. 4. New York, NY, USA: Association for Computing Machinery, aug 2000, p. 309–319. [Online]. Available: https://doi.org/10.1145/347057.347561

[18] L. Georgiadis, *Carrier-Sense Multiple Access (CSMA) Protocols*, 04 2003.

[19] "CSMA/CA: Definition and Explanation of the Method," 08 2019, [accessed 14-December-2022]. [Online]. Available: https://www.ionos.com/digitalguide/server/know-how/csmaca-carrier-sense-multiple-access-with-collision-avoidance/

[20] W. Wojdak, "Rapid Spanning Tree Protocol: A new solution from an old technology," pp. 1–5, 2003.

[21] R. Hunt and B. C. Popescu, "Comparison of PRP and HSR Networks for Protection and Control Applications," 2015.

[22] "IEEE Standard for Local and metropolitan area networks–Frame Replication and Elimination for Reliability," *IEEE Std 802.1CB-2017*, pp. 1–102, 2017.

[23] D. Ergenç and M. Fischer, "On the Reliability of IEEE 802.1CB FRER," in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, 2021, pp. 1–10.

[24] G. Noh, H. Park, H. Roh, and W. Lee, "Secure and Lightweight Subflow Establishment of Multipath-TCP," *IEEE Access*, vol. 7, pp. 1–1, 12 2019.

[25] "802.11 Reference Design: Recovery Procedures and Retransmit Limits," https://warpproject.org/trac/wiki/802.11/MAC/Lower/Retransmissions, 2014, [Online; accessed 18-December-2022].

[26] M. K. Al-Hares, P. Assimakopoulos, D. Muench, and N. J. Gomes, "Modeling Time Aware Shaping in an Ethernet Fronthaul," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, 2017, pp. 1–6.

[27] "Internet Protocol Small Computer System Interface (iSCSI) Cyclic Redundancy Check (CRC)/Checksum Considerations," Internet Requests for Comments, RFC Editor, RFC 3385, 2002. [Online]. Available: https://www.rfc-editor.org/rfc/rfc3385.txt

[28] G. Ditzel, "The Comparison/Contrast of TSN Frame Replication and Elimination for Reliability (FRER) and Parallel Redundancy Protocol (PRP)," pp. 1–13, 2020.