electronics

Review

Packet Reordering in the Era of 6G: Techniques, Challenges, and Applications

Jiaqi Lin [1], Xiaofeng Zhang [1], Xianming Gao [1], Pengtao Kang [1], Yuxi Zhou [1], Ying Ouyang [2] and Tao Feng [1,*]

[1] Institute of System Engineering AMS PLA, Beijing 100039, China; jiaqilin99@163.com (J.L.)
[2] School of Telecommunications Engineering, Xidian University, Xi'an 710126, China
* Correspondence: feng09@163.com; Tel.: +86-133-9457-3338

Abstract: The advent of sixth-generation (6G) networks brings unmatched speed, reliability, and capacity for massive connections, making it a cornerstone for revolutionary applications. One such application is in vehicular networks, which have their unique demands and complexities. Specifically, they face the complex issue of packet reordering due to the high-speed movement of vehicles and frequent switching of network connections. This paper examines the impact and causes of packet reordering, its threats to network efficiency, and potential countermeasures, particularly in the context of 6G-enabled vehicular networks. We introduce end-to-end methods and metrics to address packet reordering in 6G, discussing the development trends and application prospects. Our findings highlight the emergence of sophisticated strategies, such as prediction and avoidance, to manage packet reordering. They also reveal potential applications to boost network reliability, emulate traffic distributions, and enhance data security. Furthermore, we anticipate a growing integration of machine learning and data-driven optimization in tackling packet reordering. The insights provided aim to influence the future design and optimization of 6G networks, particularly concerning packet management and performance. This paper aims to assist researchers and practitioners in effectively leveraging packet reordering to promote efficient and secure operations of future 6G networks.

Keywords: packet reordering; 6G networks; vehicular network; packet reordering metrics

Citation: Lin, J.; Zhang, X.; Gao, X.; Kang, P.; Zhou, Y.; Ouyang, Y.; Feng, T. Packet Reordering in the Era of 6G: Techniques, Challenges, and Applications. *Electronics* **2023**, *12*, 3023. https://doi.org/10.3390/electronics12143023

Academic Editor: Djuradj Budimir

Received: 14 June 2023
Revised: 4 July 2023
Accepted: 7 July 2023
Published: 10 July 2023

1. Introduction

In our rapidly evolving age of information technology, mobile communication networks have upgraded to infrastructures vital to social and industrial functions [1]. From rudimentary first-generation (1G) mobile network technologies to sophisticated fifth-generation (5G) broadband cellular network technologies, each technological iteration endows users with faster data transmission rates, manages more robust connections, and extends the range of application scenarios. However, despite the current power of 5G networks, they still reveal the limitations of our existing network technology. These limitations manifest as restrictions on spectral resources, congestion, and fluctuations in communication quality. These challenges become more prominent when considering the requirements for massive concurrent connections, low-latency communication, and big data processing. To meet the stringent demands of our future intelligent internet—an internet envisioned as a harmonious combination of the Internet of Things and artificial intelligence—researching and designing the next generation of mobile communication network technology becomes critical [2]. Against this backdrop, the dawn of the sixth-generation (6G) mobile communication network begins to break, illuminating the direction for future communication technology.

Considering the development trajectory of mobile networks, we expect the emerging sixth-generation (6G) network to preserve the essence of the existing 5G architecture and further enhance it. This could involve introducing more licensed frequency bands and turning to a more dispersed network architecture. Based on historical precedents of past commercial wireless communication systems and emerging expectations for 6G, we can predict that it might begin commercial roll-out within the next decade [3]. In the realm of 6G networks, a suite of characteristics, including astonishing speed, unmatched reliability, minimal latency, and the ability to bear massive connections, will infuse life into emerging application scenarios. This could potentially realize visions of precision medicine, intelligent disaster prediction, hyper-realistic virtual reality, smart cities, and automated industries [4].

One specific application scenario, that is, vehicular networks, plays a pivotal role in the context of 6G applications. With the emergence of applications such as autonomous driving, traffic management, and in-car entertainment services, the demand for low-latency communication and large-scale concurrent connections becomes increasingly intense. Additionally, due to the high-speed movement of vehicles and frequent switching of network connections, vehicular networks introduce unique challenges to the issue of packet reordering. As such, studying packet reordering issues in vehicular networks within a 6G environment becomes particularly crucial and complex.

Moreover, these challenges extend beyond vehicular networks to encompass the broader 6G network environment. As we confront larger scale concurrent connections, lower latency requirements, and the need to handle more data, the issue of packet reordering emerges as a significant concern. This phenomenon, triggered by the labyrinthine transmission conditions of complex networks, disrupts latency times and scrambles the order of packet arrivals. Notably, packets belonging to one flow may be blocked by packets from multiple other flows, leading to cache losses and a surge in the number of CPU instructions. Consequently, this reduces the efficiency of packet processing applications, thereby threatening the reliability and efficiency of the broader 6G network [5]. In conclusion, the evolution of mobile communication technology, specifically towards 6G, carries the promise of transforming our ways of life, work, and entertainment. However, this technological leap comes with new challenges. Among these, the issue of packet reordering stands out as a key challenge that needs to be addressed to unlock the full potential of 6G networks. We can understand the importance of this issue from several aspects:

- Network Complexity. As 6G extends its tentacles, packets may traverse many paths during transmission. This is a departure from the previously predictable single-path routing. Unfortunately, this complexity opens the door to packet reordering, as bandwidth, loss, and latency parameters may vary among the multiple links in multipath routing. The chain reaction of packet reordering goes beyond merely a scrambled order. It may cause severe buffer overflow at the receiving end and trigger unnecessary packet loss and congestion invalidation of retransmissions. This undoubtedly increases network load, fails to maximize link utilization, and reduces the overall transmission performance of network coupling.
- Low Latency. Moving into 6G causes an increasing demand for low-latency communication. As we stand at the edge of emerging application scenarios such as virtual reality and autonomous driving, the demand for precise latency control becomes more pressing than ever. It should be understood that the issue of packet disarray could potentially affect overall network performance by increasing communication latency.
- Massive Concurrent Connections. In the realm of 6G networks, the ability to support a large number of concurrent device connections is not just an advantage, but a necessity. Unfortunately, this necessity could potentially exacerbate the issue of packet disarray. Congestion caused by packet reordering may severely impact the user experience and disrupt smooth business operations. Therefore, solving the packet reordering problem is key to ensuring the efficient operation of 6G networks.

While the problem of packet reordering has been studied in the context of earlier network generations, there is still much to understand about how it manifests and impacts performance in the emergent 6G networks, especially in the context of vehicular applications. As pointed out in recent studies [6], packet reordering in 6G environments presents unique challenges due to the extremely high data rates, ultra-reliability, ultra-low latency, and large number of devices involved. Thus, the need for research in this area has been recognized by the scientific community, as evidenced by the increasing number of publications on the topic [7]. Further research into the relationship between packet reordering and 6G is of practical significance as it explores the impact, causes, and solutions of this topic. The objective of this paper is to investigate the impact of packet reordering on networks, uncover its root causes, explore the potential risks it poses to user experience and network services, and offer theoretical insights and technical guidance for 6G networks. In addition, we introduce several end-to-end methods and metrics to guide packet reordering in the context of 6G. Lastly, we discuss the application prospects and development trends of packet reordering in these advanced vehicular network environments powered by 6G technology.

2. Packet Reordering

2.1. Overview

Packet transmission is a fundamental method of modern network communication. It involves dividing data into packets, encapsulating them, and transmitting them from the data source to the receiving end over single or multiple links, following specific protocols. However, in the actual data path, packets may experience disorder, meaning that the order of packets received by the receiver does not match the order in which they were sent by the sender due to various factors during the data processing [8]. Let us consider two concurrent links as depicted in Figure 1. Assume that the delay of the two links satisfies $D_1 \ll D_2$ packets with serial numbers 1, 4, and 6 from the first path arriving at the receive buffer first in sequential order. Subsequently, packets 5, 3, and 2 from the second path arrive. As packet 2 is awaited, packets 1, 4, 6, 5, and 3 remain in the buffer, causing buffer congestion and degrading network performance. This out-of-order phenomenon indicates a discrepancy between the data flows of senders and receivers. Furthermore, this discrepancy occurs because the sequence numbers of the packets do not match the receive index. When packets with larger numbers arrive at the device before those with smaller numbers, the earlier-arriving packets are cached and wait for the arrival of packets with smaller numbers. Ultimately, these packets are reordered together before being delivered to upper-layer applications.

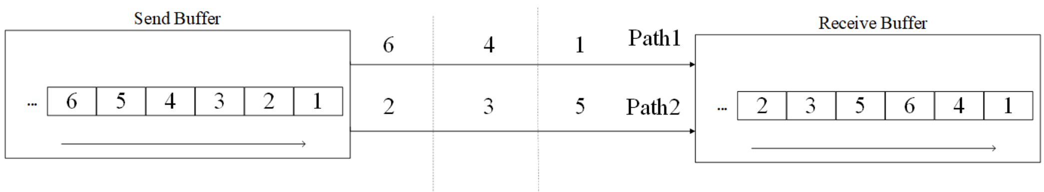

Figure 1. An example of packet reordering.

Packet reordering is a well-known phenomenon on the Internet, referring to the change in relative order of certain packets within a data flow during network transmission, and the behavior of restoring the original order of packets at the sender through the analysis of packet relationships [9]. Discussions on packet reordering in the industry can be categorized into two types [10], both originating from the 1990s. The first type focuses on measurement methods and experimental research, addressing two main aspects: (1) Measurement environment, including the selection of observation points and datasets. Active probing with specific types of packets (such as TCP or ICMP) is required for measurement at the sender/receiver, while passive detection can be performed at certain points in the backbone network. (2) Reordering determination algorithms, including the definition and classification of reordering, as well as the rules for determining reordering. The second type involves specialized research on reordering, covering aspects such as causes, measurement techniques, evaluation methods, and improvement approaches. In 1999, Paxson et al. [11] undertook two extensive sets of end-to-end experiments involving 35 Internet sites, each consisting of 20,000 TCP bulk transfers. The findings from these experiments revealed the existence of asymmetric Internet paths that led to anomalous network events, such as out-of-order delivery. Interestingly, the frequency of these anomalies was markedly different between the two studies, with the first experiment identifying these events in 2% of cases, while the second experiment observed them at a lower rate of 0.26%. Subsequently, Loguinov and Radha [12] conducted early measurement research on UDP flows, and Gharai [13] conducted research on streaming traffic. After the two measurement studies in 1999, Bennett et al. [14] initiated the second type of research, proposing that packet reordering on the Internet is a pathological network behavior caused by the parallelism of Internet components and links. Large-scale packet reordering on the network can significantly degrade TCP performance, so it should be avoided as much as possible or quickly recovered from. This led to a series of discussions and research on packet reordering in the industry.

In order to gain a deeper understanding of the research status in the field of packet reordering, we conducted a comprehensive literature analysis. We selected 728 articles related to packet reordering published between 1995 and 2022 from the Web of Science (WOS) database. The selection criteria were based on the relevance of the articles to packet reordering, without restricting us to any particular network type or application scenario. We used CiteSpace, a software tool for scientific literature analysis, to analyze the selected articles from three aspects: keyword clustering, citation clustering, and keyword timeline. The keyword clustering helped us identify the central themes and recurring topics within the body of literature. The citation clustering allowed us to identify the most influential works and authors in the field. Through the keyword timeline, we traced the evolution of research focus over time, revealing trends, shifts, and gaps in the existing literature. This systematic approach ensured that our analysis captured the breadth and depth of research on packet reordering in different network contexts, laying a solid foundation for the subsequent focus on 6G networks and vehicular applications.

As shown in Figure 2, after analyzing the citation clustering data, it was found that early research primarily focused on packet scheduling, TCP protocols, and load balancing. With the development of network technology, subsequent research gradually shifted towards emerging fields such as heterogeneous wireless networks, network traffic watermarking, and 5G networks. Figure 3 reveals the citation clustering status in the field, with the largest number of nodes found in cluster 0, reaching 26, and a homogeneity value of 0.952. Topics such as load distribution, concurrent multipath transfer, multipath forwarding, and packet delay variation are closely associated. Although the average year is relatively early (2008), these topics remain the focus of discussion in the field. Clusters 6, 7, and 10 have average years after 2016, and they focus more on multiple paths, data center networks, and bandwidth efficiency. This indicates that the literature in the field continues to adapt to emerging scenarios based on the classic foundation, highlighting the continuous attention given to packet reordering by researchers.

In Figure 4, based on the average year of clustering, we can observe fluctuations in research trends among these subfields during different time periods, with recent years witnessing a growing focus on 5G network research. Furthermore, we can observe the interconnections between these subfields. For example, the issue of packet reordering is relevant in the fields of TCP protocols, heterogeneous wireless networks, and 5G networks. This demonstrates that packet reordering has a broad impact across different network environments and application scenarios.

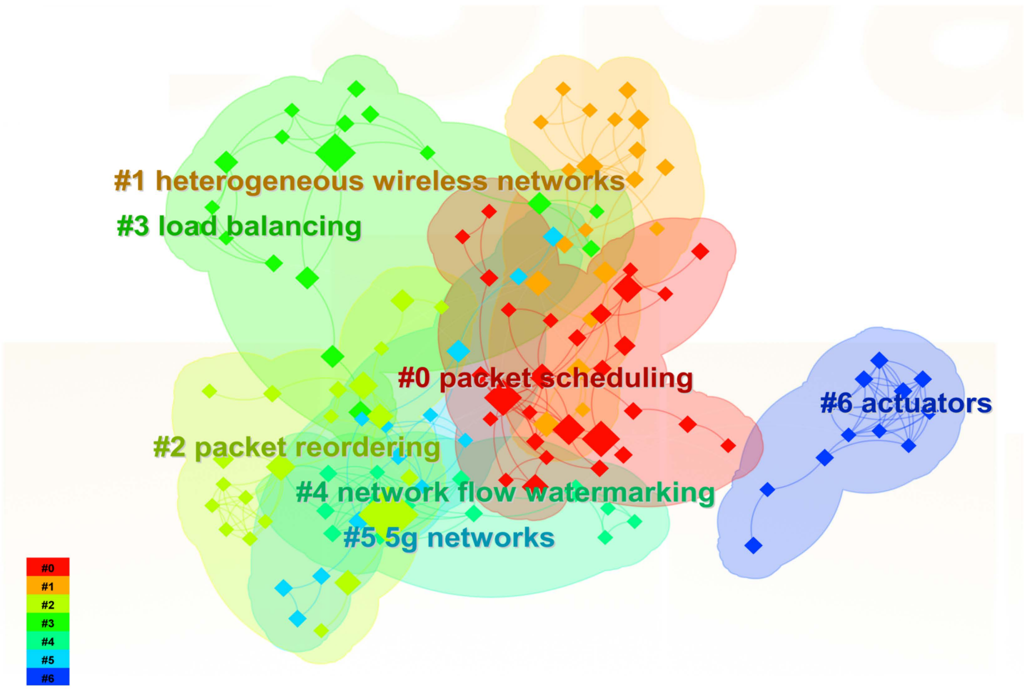

Figure 2. Keyword clustering analysis of the literature.

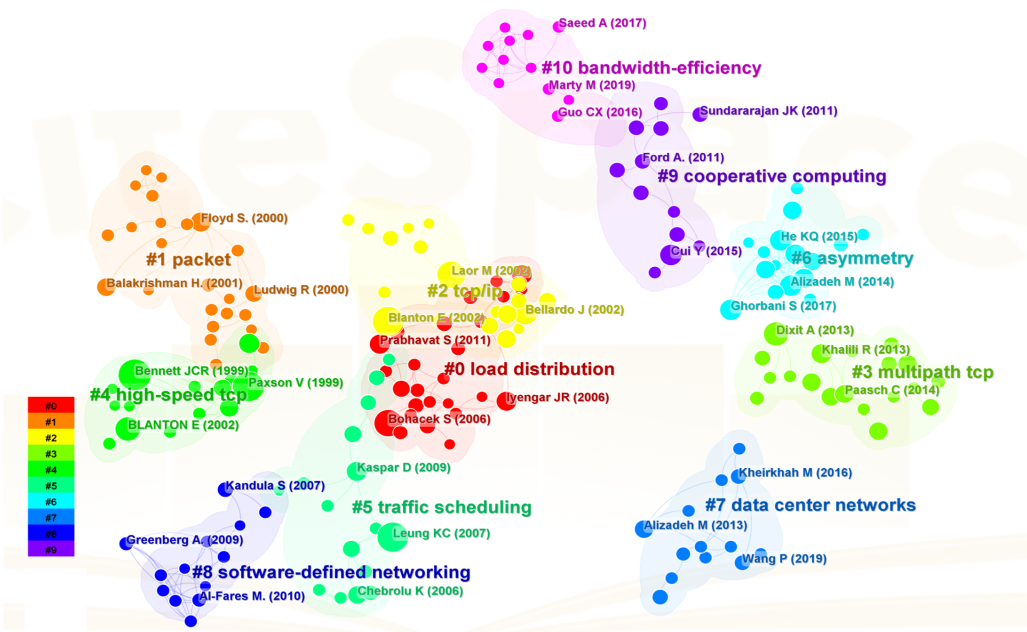

Figure 3. The clustering analysis of related literature. The data is based on the research findings of various authors.

When analyzing the literature in the field of packet reordering, we can observe that this field has gradually gained attention since the late 1990s and covers multiple subfields, such as load balancing, TCP protocols, network performance, and wireless networks. As research progresses, the number of publications in this field has increased over the years, flourishing in the early 21st century. An examination of the academic literature demonstrates a significant rise in research interest regarding packet reordering over the past two decades. By searching the term "packet reordering" on the dblp computer science bibliography, it was revealed that since the release of the first seminal paper in 1999, the number of related publications has seen substantial growth. Specifically, the period from 2001 to 2010 saw 57 papers on the topic, while the following decade from 2011 to 2020 recorded 39 more. This upward trend underscores the escalating attention this issue is garnering within the realm of network technology development. Many scholars generally agree that packet reordering is a common and challenging anomaly that is difficult to avoid in various network environments. These viewpoints include the research by Bennett, Partridge, and Shectman [14], who argue that packet reordering on the Internet is unavoidable and has a significant impact on network performance. Their work is highly influential and widely referenced in this field. Additionally, the research by Bohacek et al. [15] suggests that packet reordering is actually a normal part of the operation for many routers that involve parallel paths through switches. Due to the scheduling algorithms used and variations in packet sizes and arrival times, packets entering a single interface may be reordered. Their work serves as a bridge in the citation network, connecting many other related studies and providing valuable insights for researchers. While some scholars argue that the probability of packet reordering occurrence is low, they also acknowledge that this phenomenon can impact network performance to varying degrees.

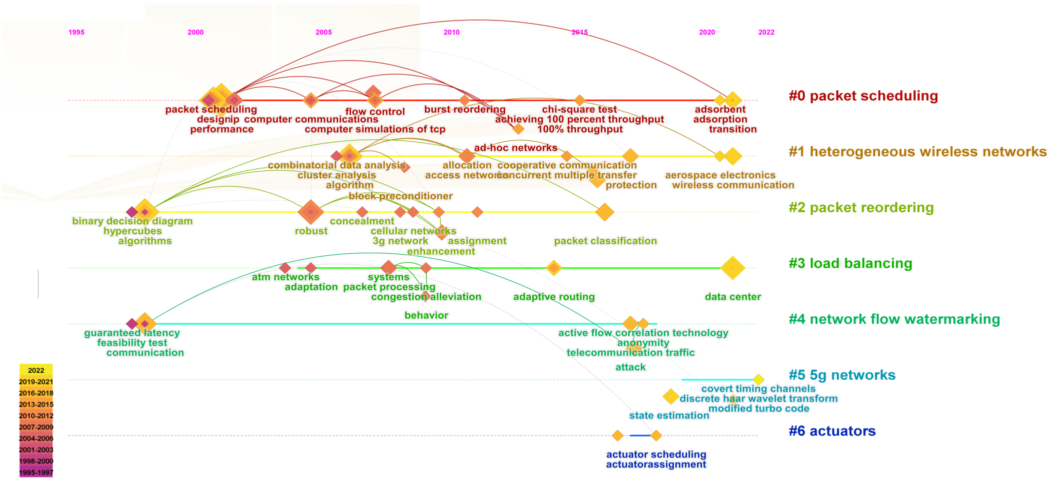

Figure 4. Keyword timeline analysis.

2.2. The Causes of Packet Reordering

The causes of packet reordering are multifaceted and can be analyzed from several aspects, including network structure and topology, packet transmission protocols and technologies, network devices and resource scheduling strategies, network congestion, and traffic control, as well as heterogeneous network environments and application scenarios. Understanding and studying the causes of packet reordering contribute to improving network transmission efficiency and stability, reducing network latency, and minimizing packet loss rates.

2.2.1. Network Structure and Topology

In complex network environments, packets may traverse different network nodes and links during transmission. The structure, performance, and connectivity of these nodes and links can lead to packet reordering.

1. Path Asymmetry. Multipath transmission, compared to single-path transmission, can balance the load and provide better congestion performance [16,17]. However, in packet-level multipath routing, the differences in path attributes can cause packet reordering during transmission. Path attributes that have been proven to affect packet reordering include, but are not limited to, sending rate, propagation rate, link bandwidth, packet loss rate, delays (processing delay, queuing delay, transmission delay, propagation delay, tail delay), packet intervals, and packet sizes [13,18], as shown in Table 1.

Table 1. Explanation of path attributes related to packet reordering.

Path Properties	Description or Explanation
Sending Rate	The rate at which hosts or routers send data onto the digital channel. TCP flows with higher sending rates are more likely to experience TCP fast retransmission and recovery with more errors [19].
Propagation Rate	The speed at which electromagnetic waves propagate in the channel.
Bandwidth	The amount of data that can be transmitted in a network per unit of time.
Packet Loss Rate	The ratio of the number of lost packets to the total number of packets sent.
Processing Delay	The time it takes for a host or router to process a packet upon receiving it, including tasks such as splitting, encapsulation, analysis, and computation.
Queuing Delay	The time that packets spend in input and output queues waiting to be processed within a router. Although queuing delay is generally not considered significant, its impact on packet reordering cannot be ignored [20].
Transmission Delay	The time it takes for a host or router to send a data frame, i.e., the time from when the first byte is sent to when the last byte leaves the machine.
Propagation Delay	The time it takes for electromagnetic waves to propagate a certain distance in the channel, occurring in both the sending device and the external transmission media.
Tail Delay	Furthermore, known as high percentile delay, it refers to the high delay that clients rarely experience. There are several factors contributing to tail delay, including contention, garbage collection, packet loss, host failures, and strange operations performed by the operating system in the background. Tail delay can also affect the delay in packet reordering and the number of out-of-order packets received [21].
Inter-Packet Spacing	The time interval between packets. There is a strong correlation between inter-packet spacing and packet reordering [13,22]. Smaller inter-packet spacing may increase the probability of packet reordering. In other words, for the same packet size, a higher sending rate increases the likelihood of packet reordering [19].
Packet Size	The size of packets when they need to be fragmented if smaller than the MTU. Smaller packets are more prone to frequent packet reordering compared to larger packets [18].

2. Path Changes. In the absence of packet loss, packets on a single path will not experience reordering. However, when encountering heavy load, instability, or failure on a link, or when choosing a more optimal path due to latency and congestion, packets may oscillate between available routes to the destination, leading to different delays and packet reordering at the destination [23,24]. For example, in mobile ad hoc networks, they exhibit dynamic topological characteristics, where nodes can move freely, resulting in changes in the network's topology.
3. Limited Link Bandwidth and Time-Varying Capacity. Due to the dynamic changes in the network topology, the amount of traffic forwarded by each node to destinations other than itself also changes over time. Therefore, unlike wired networks, the capacity of links in wireless networks exhibits time-varying characteristics.

2.2.2. Packet Transport Protocols and Technologies

1. Transport Layer Protocols. Transport layer protocols such as TCP and UDP have different characteristics when handling packets. TCP is reliable and ensures ordered delivery by retransmitting lost packets [25] and sequencing received packets. On the other hand, UDP is a connectionless protocol that is more sensitive to packet ordering. It does not guarantee the ordered delivery of packets [26]. Therefore, when using UDP, the phenomenon of packet reordering is more noticeable and has a deeper impact.

2. Multipath Transport Protocols. Examples include Multipath TCP (MPTCP) and Multipath UDP (MPUDP). These protocols allow packets to be transmitted through multiple paths, thereby improving network resource utilization and enhancing network robustness. If packets are forwarded through multiple paths, the introduction of asymmetric path characteristics is expected to introduce more parallelism and result in a higher occurrence of packet reordering.
3. Link Layer Technologies and Routing Protocols. Data link layer technologies (such as Ethernet, ATM, Frame Relay, etc.) and routing protocols (such as OSPF, BGP, RIP, etc.) can impact the transmission rate, delay, path, and devices traversed by packets, thus affecting the packet arrival order.

2.2.3. Network Devices and Resource Scheduling Strategies

1. Router Internal Parallelism, Reordering, and Forwarding Delays. To achieve better I/O performance and increase throughput, modern routers support packet striping [14,27]. Although many network processors have internal hardware to track traffic and reduce packet reordering within the router, multiple parallel links are still used to connect to the next-hop router, especially in load-balanced switches. Continuous packets on an input interface propagate to all intermediate ports, encountering different queuing delays [28] or varying queue lengths, resulting in different transmission times and inconsistent ordering between transmission and reception, leading to packet reordering. Additionally, in complex network environments, interoperability is required among different vendors and types of network devices. These devices may have differences in packet processing, such as different queuing strategies and caching mechanisms, resulting in packet reordering. FPGA (Field Programmable Gate Array) is gradually replacing ASIC (Application-Specific Integrated Circuit) due to its shorter development cycle, programmability, and high flexibility. However, low-cost ASICs are still used in some network domains, and packets processed by different circuitry may be reordered. During route updates, routers may pause forwarding buffered packets to handle the route updates, causing newly arrived packets to be held back and resulting in packet reordering [25].
2. Resource Scheduling Strategies. Modern network processors need to support a rich set of services. For example, a multiservice edge router may require support for encryption, decryption, firewall, intrusion detection, and many other services. Packet processing cores used in these processors are often small, and if cores and caches are allocated arbitrarily, it can lead to performance degradation for latency-sensitive network processors, such as packet loss or out-of-order packet transmission. Load balancers also migrate some traffic from overloaded cores to underutilized cores. However, flow migration is undesirable, as incoming packets may experience less queuing delay compared to old packets waiting in the overloaded core queue. This leads to poor data locality and packet reordering.
3. Protocol Specifications for Network Devices. In IPv6, the average packet reordering rate is much lower than in IPv4 networks for two reasons [29]: (1) IPv6 discourages fragmentation in most cases, while in IPv4, hosts and routers can perform packet fragmentation; (2) IPv6 simplifies the basic header, speeding up packet processing and improving the efficiency of packet handling.

2.2.4. Network Congestion and Traffic Control

Network congestion, traffic control, and the application of different protocols and technologies all have an impact on the packet transmission order in a network. These factors interact with each other, exacerbating the complexity and diversity of packet reordering phenomena. In terms of network congestion and traffic control, congestion can lead to reduced packet transmission rates, increased latency, and higher packet loss rates. Traffic control and congestion control strategies adjust the sending rate, discard policies, and route selection to reduce network congestion. However, these strategies may cause fluctuations in the packet transmission rates in the network, thereby affecting the arrival order of packets. Additionally, different queue management algorithms used in network devices to queue and schedule packets can also have varying effects on the packet transmission order.

2.2.5. Heterogeneous Network Environment and Application Scenarios

In heterogeneous network environments and application scenarios, the phenomenon of packet reordering exhibits higher complexity. These scenarios include wireless networks, mobile networks, data center networks, cloud computing environments, software-defined networks (SDN), and network function virtualization (NFV) [30]. The diverse network structures, devices, protocols, and technologies in these scenarios collectively impact the transmission order of packets. For example, factors such as channel quality, mobility, and multipath propagation in wireless and mobile networks result in fluctuations in packet transmission rates and delays. In data center networks and cloud computing environments, load balancing strategies, virtual machine migration, and dynamic resource allocation techniques and policies lead to changes in the transmission paths and rates of packets. Similarly, in SDN and NFV environments, the dynamic scheduling of network resources and functions, virtualization technologies, and flexible network programming capabilities collectively influence the transmission order of packets. Research based on actual measurements indicates that packet reordering in heterogeneous networks has surpassed that caused by regular connections in high-speed wide-area networks [31–33].

In addition to the primary reasons mentioned above, factors such as path count [30], bursty traffic [34], and differentiated services handling for flows violating quality constraints can also contribute to the occurrence of packet reordering. In real network environments, these factors interact and collectively influence the transmission order of packets. Understanding these reasons is crucial for studying the phenomenon of packet reordering and its impacts.

2.3. The Impact of Packet Reordering

Data packet reordering has various effects on network communication, primarily involving application performance and user experience, as well as network security and performance diagnostics. We now discuss the specific reasons and outcomes for each of these aspects.

2.3.1. Application Performance and User Experience

Packet reordering can lead to increased latency, decreased throughput, and jitter in application-layer data transmission, thereby impacting the Quality of Service (QoS) and Quality of Experience (QoE) for applications. In delay-sensitive applications based on UDP, such as multimedia software, receiving out-of-order packets after the playback time is as detrimental as lost packets. The cost of recovering from packet reordering at the user end is high, as it increases the buffer requirements and introduces processing-related delays. Severe packet reordering can cause a buffer overflow, leading to the dropping of a large number of out-of-order packets, resulting in issues such as audio–video stuttering and blurry visuals, ultimately degrading the user experience.

In the case of reliable transport protocols such as TCP, the impact of reordering depends on whether the packets are reordered in the forward path (forward path reordering) or the acknowledgments are reordered in the reverse path (reverse path reordering) [35]. As shown in Figure 5, in the forward path from the sender to the receiver, forward path reordering refers to the arrival of packets out of order, which can be perceived as packet loss. In the reverse path, reverse path reordering occurs when ACKs arrive out of order at the destination, leading to the loss of TCP's ACK self-clocking. This affects the ACK sequence from the receiver to the sender, resulting in burst packet transmissions and congestion. Both types of reordering significantly impact the receiver's efficiency and TCP end-to-end performance, leading to a reduced congestion window and unnecessary retransmissions [36]. We next explain in detail several specific impacts.

1. Spurious Retransmissions. TCP has two methods to trigger its retransmission mechanism [37]. The first method relies on the reception of duplicate ACKs, indicating that the receiver has lost some data [38]. After receiving a required number of consecutive duplicate ACKs (usually three), the TCP sender retransmits the first unacknowledged segment [39] using the fast retransmit and recovery algorithm. The second method involves the TCP sender maintaining a retransmission timer. If a segment remains unacknowledged before the retransmission timeout (RTO) expires, the timer triggers the retransmission of the segment. Upon a retransmission timeout, the TCP sender enters RTO recovery, where the congestion window is initialized to one segment, and the unacknowledged segments are retransmitted using the slow-start algorithm. The retransmission timer is dynamically adjusted based on the measured round-trip time (RTT) [40].
   According to RFC4138 [41], spurious retransmission refers to cases where a retransmission appears to be a timeout but is not an actual timeout. There are several reasons for spurious retransmissions:
   - In some mobile networks, network latency may spike during network handovers.
   - When the available bandwidth in the network suddenly decreases, the network RTT can experience a sudden increase, leading to the estimation of an erroneous RTO (RTO is determined by the sum of a smoothed round-trip time-weighted moving average and a multiple of the average deviation between the RTT and the smoothed average [40]).
   - Packet loss in the network can cause spurious retransmission. When the sender receives three consecutive duplicate ACKs, reordered packets may be mistaken for lost packets, triggering a series of actions in the protocol stack. Persistent and significant packet reordering often results in some TCP segments being unnecessarily retransmitted, wasting bandwidth. These packets have actually been successfully received, but due to the misordering, the sender mistakenly assumes they are lost, thereby reducing the efficiency of data transmission and potentially leading to congestion collapse [42].
2. Congestion window reduction. TCP is unable to distinguish between packet reordering and packet loss. The receiver of TCP expects that packets from the same data stream are consecutively numbered. After receiving several consecutive duplicate ACKs, the sender may assume that a particular packet is lost. Consequently, it will initiate retransmission and recovery algorithms, leading to a multiplicative reduction in the congestion window size (cwnd) of TCP, transitioning from a "slow start" to a gradual increase in transmission speed. In networks where packet reordering persists and is substantial, TCP will erroneously retransmit data segments, keeping its cwnd unnecessarily small. This can cause the receiver to be uncertain whether an ACK received is for the first transmission of a segment or for retransmission. RTT samples may be discarded, and both RTT and RTO can be underestimated, limiting the transmission speed of TCP and severely impacting its performance [43]. It is worth mentioning that, as described in Section 2.2, in some studies, a reduction in the congestion window is also considered a result of packet reordering.
3. ACK clock interruption [44]. TCP nodes are distributed worldwide, making it impossible to achieve global clock synchronization for driving cooperative network behavior. Therefore, TCP relies on ACKs and timeout timers to achieve this synchronization. Ideally, with a stable ACK clock, the TCP sender would continuously feed the data stream into the network driven by that clock. However, in the case of reverse path reordering, the arrival of out-of-order packets disrupts the sequence, causing the source node to send multiple packets. This situation interrupts the ACK clock and results in more bursty TCP transmissions. These bursty transmissions can lead to network congestion and even congestion collapse [42], as the network may struggle to handle the sudden increase in data traffic.

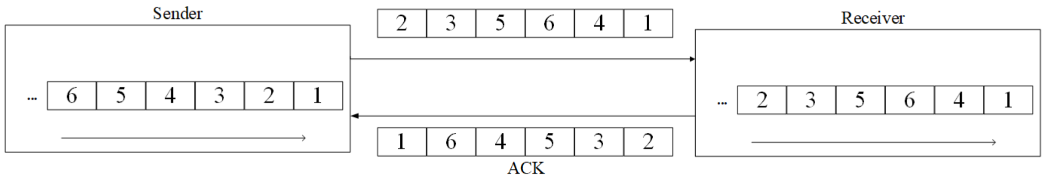

Figure 5. Forward path reordering and reverse path reordering.

2.3.2. Network Security and Performance Diagnosis

Packet reordering has significant implications for network stability and security. Even a small fraction of reordered packets in the backbone link can cause a significant decrease in throughput [45]. Packet reordering can mislead certain protocols (e.g., DCCP, SCTP) and security-related functionalities (e.g., network intrusion detection and prevention systems), leading them to incorrectly perceive network issues or attacks. For example, intrusion detection and defense systems may mistakenly interpret packet reordering as a sign of an attack, resulting in false positives. These systems need to maintain the state of each flow [46–48]. Additionally, attackers may exploit packet reordering to obfuscate these systems and achieve malicious objectives. Packet reordering can also impact the performance of the anti-replay sliding window mechanism in IPSec [35], causing legitimate packets to be mistakenly identified as duplicate attacks and rejected [49]. Unlike packet loss, accurately identifying and pinpointing the root causes of packet reordering poses significant challenges for network monitoring and management personnel. Therefore, network administrators require higher skills and more sophisticated tools for network management and troubleshooting to ensure stable and secure network communication.

Furthermore, packet reordering is contagious [20]. This means that the reordering of one packet can trigger delays for the same data flow or even other data flows, leading to suboptimal data transmission on the link and expanding the scope of impact. The coupled overall transmission performance is reduced. Finally, it is worth mentioning that packet reordering is not entirely disadvantageous. For instance, it can be deliberately induced and used as a means to enhance channel reliability, simulate realistic traffic distributions, augment stealthiness, and enable more secure and private data transmission [50]. More advanced applications of this nature warrant further investigation, such as the software solution Reframer [5], which intentionally delays and reorders packets to augment traffic locality. Remarkably, this approach boosted the throughput of network service chains by 84%, reduced the flow completion time of web servers by 11%, and improved their throughput by 20%.

3. The Solution to Packet Reordering

In network communication, packet reordering is a common and challenging issue. Due to the complexity and uncertainty of networks, when packets arrive at the destination out of order, the receiver may require additional processing steps and resources to reorder the packets, resulting in processing delays and performance degradation. In some cases, if packet reordering is severe, the receiver may struggle to correctly reconstruct the original data, leading to data corruption or loss. Packet reordering not only affects data integrity but can also cause a decline in network performance. Therefore, addressing the problem of packet reordering is crucial for improving network performance.

Within the network protocol stack, the transport layer plays a critical role as it is responsible for end-to-end data transmission, including the management of packet ordering. As shown in Figure 6, the process involves the packets being sent from the sender to the path, experiencing reordering along the path, and then being reordered in the receiver's buffer. Current solutions focus on these three stages, aiming to predict and avoid reordering before it happens, tolerate or identify it when it occurs, and initiate fast recovery mechanisms. Solutions for packet reordering primarily reside in the transport layer, and this chapter focuses on discussing packet reordering solutions at this layer. These solutions include prediction, avoidance, identification, and tolerance of packet reordering, which can assist network applications in handling packet reordering more efficiently and reliably.

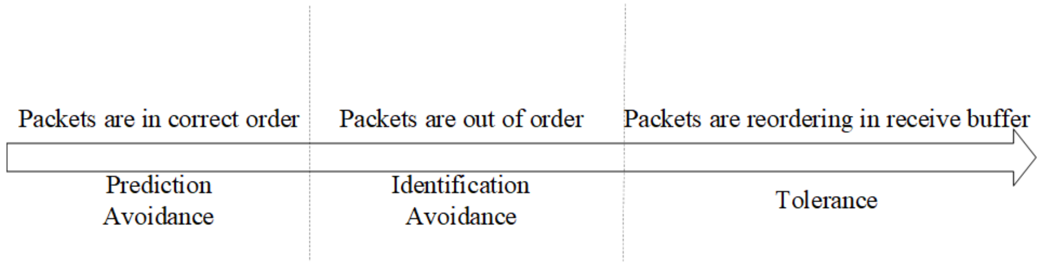

Figure 6. Response methods starting from each phase of packet reordering.

3.1. Packet Reordering Prediction

When we talk about packet reordering prediction, we refer to mechanisms or algorithms that attempt to predict potential packet reordering in the network and adjust the sending or receiving strategies accordingly. We can approach this from the following perspectives:

3.1.1. Path-Based Prediction

In multipath transmission, the arrival order of packets can be influenced by the attributes of multiple paths, such as delay and packet loss rate. It is not always ideal to have more parallel paths, as the aggregated throughput can be limited by high-latency transmission links. One approach is to consider these path attributes before transmission and build a reordering model based on the network state and optimization criteria [51–53]. This model can select the optimal transmission path, such as choosing the path with the lowest utilization [54] or selecting high-performance paths as the primary paths while using others as backup paths for retransmission and redundancy [55]. The authors [56] propose a route-aware protocol to calculate the trust value of each node and then select the optimal path for transmission.

3.1.2. Importance-Based Prediction

In addition to path attributes, we can also consider the importance of each packet. For example, certain packets may contain critical information for the entire information flow, and we may need to allocate better paths for these packets or use redundancy transmission to ensure their arrival [53], enabling subsequent batch decisions.

3.1.3. Redundant Transmission Based on Network Coding

To ensure data integrity and reliability, we can use a technique called network coding, which adds redundant information to improve the reliability of packets [55,57,58]. As shown in Figure 7, network coding primarily focuses on basic linear computations, enabling each packet to be replaced by other packets, thus enhancing the overall robustness of the network. This is particularly useful for rare events with high queuing delays. The latest method named Coding-based Distributed Congestion-aware Packet Spraying mechanism or CDCPS [59] eliminates out-of-order packets completely and effectively reduces the average and 99th flow completion time by up to 73% and 78%, respectively, over the state-of-the-art load balancing scheme.

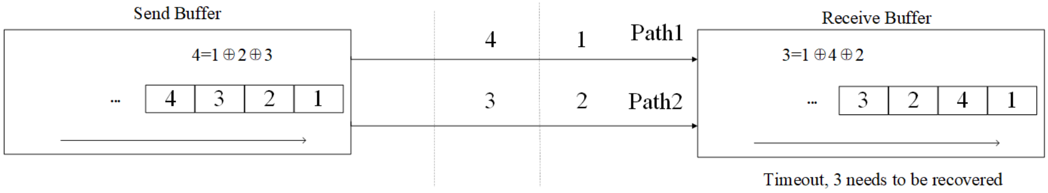

Figure 7. An example of network coding.

3.1.4. Prediction Based on Stochastic Compensation Effect

In networks, path attributes may dynamically change over time. Most attribute-based calculations assume deterministic proportional scheduling. However, there is no evidence that deterministic scheduling is effective in highly jittery networks where path attributes are random. In such stochastic systems, using a stochastic strategy instead of a deterministic one may be more efficient. This involves partially compensating for the variations in system state parameters through random input parameters, known as the stochastic compensation effect [54].

3.1.5. Prediction Based on Reverse Engineering

Some studies propose reverse engineering methods that adjust the sending packet strategies based on predicting the receiver's state. This approach aims to ensure that packets arrive at the receiver in the correct order.

3.2. Packet Reordering Avoidance

3.2.1. Prediction Based on Packet-Level Traffic Allocation

In addition to proactive prediction, another approach to mitigate packet reordering is to distribute packets of a flow across multiple paths using traffic splitting or transmission control strategies. This strategy aims to improve throughput and reduce end-to-end latency. However, if the path attributes are asymmetric, it may increase the likelihood of reordering. To address this issue, a solution called QDAPS (Queueing Delay Aware Packet Spraying) has been proposed. QDAPS splits a single flow into packets and then re-routes each packet at a fine granularity by carefully selecting output ports to mitigate packet reordering [60].

3.2.2. Prediction Based on Flow-Level Traffic Allocation

This strategy considers the attributes of the entire flow, such as real-time performance of links, load balancing, and ordered packet transmission. Based on this information, a data allocation scheduling model is constructed, and strategies are implemented to improve congestion conditions of subflows, reduce performance differences between concurrent links, minimize packet reordering, and prevent receiver buffer congestion [61]. Strategies at the flow level include Direct Hashing (DH), Table-based Hashing (TH), Highest Random Weight (HRW), etc. These strategies ensure that packets of the same flow are always mapped to the same path through hash calculations. Table-based Hash Redistribution (THR) methods attempt to improve the hash scheme by addressing the issue of unequal traffic sizes. They allocate a unique mapping table for each service, use incremental hashing for scheduling, identify the highest data rate flow, and dynamically migrate scheduling. This approach effectively avoids packet reordering while maintaining synchronization between flow positions and cache positions, leading to improved network performance [62]. Another study proposes a method that sets buffers at the output ports of switches and adopts sorting strategies for per-hash groups. This strategy implements a more flexible yet effective packet ordering rule, significantly reducing average packet delay. Moreover, this method has relatively low implementation complexity, making it more feasible for deployment and use in practical networks [28].

3.2.3. Load Balancing

Load balancing is a network management technique that evenly distributes network traffic across multiple paths or servers to prevent network congestion and optimize network performance [17]. It is a powerful tool to prevent server overload, reduce response times, and improve overall network efficiency and availability. Ideally, load balancing ensures that the actual load on each path matches the desired load, meaning the traffic carried by each path aligns with its capacity. This prevents some paths from being overloaded while others have wasted resources. However, achieving this ideal load balancing can be challenging in practice, as network traffic and load conditions may vary over time. Load balancing is not always entirely harmless, as it may interact negatively with TCP's congestion control mechanisms. TCP's congestion control is a reactive strategy that reduces data transmission rates to alleviate network load during congestion. However, if load balancing is used in such situations, traffic may be shifted to non-congested paths, thereby inadvertently increasing network load and potentially worsening the congestion situation [63].

3.2.4. Flow Partitioning

To better balance the load on all paths and mitigate bias, flow partitioning strategies have been explored. Methods such as MATE [64], FSLB [65], TeXCP [66], COPE [67], and others employ this strategy in practice, involving dynamic approaches for flow splitting performance. The Flowlet-based Router Engine (FLARE) [68] is an important example that splits a data flow into multiple subflows based on traffic distribution policies. These subflows are guided to switch to paths with the lowest utilization. If the inter-arrival time of two consecutive packets in the same data flow exceeds the maximum time required to send packets via parallel paths, the likelihood of packet reordering decreases. However, burstiness of the data flow may result in shorter inter-packet intervals, affecting the performance of FLARE [34]. However, the latest approaches find that the above methods reduce link utilization due to their inflexibility. The authors of [60] proposed a Queuing Delay Aware Packet Spraying (QDAPS), that effectively mitigates the packet reordering for a packet-level load balancer and reduces the process completion time (FCT) by 30%–50% over state-of-the-art load balancing mechanisms.

3.2.5. Flow Truncation Load Balancing Based on Continuous Inter-Packet Arrival Time

This approach, proposed by FCLB (Flow Chopped Load Balancing algorithm) [17], divides data flows into multiple segments based on the continuous inter-packet arrival time. This ensures that packet chopping on each path approaches the desired chopping rate. This method effectively improves packet ordering and enhances network performance.

3.3. Packet Reordering Identification

Packet reordering identification techniques refer to the use of certain methods at the receiver side, such as examining packet sequence numbers or timestamps, to determine if packet reordering has occurred. This is crucial for many packet-order-dependent network applications, such as video streaming or VoIP, as packet reordering can lead to performance degradation in these applications. Additionally, for protocols that require maintaining packet order, such as TCP, accurate identification of packet reordering is essential as it helps optimize congestion control strategies and improve network performance.

3.3.1. Acknowledgment Mechanisms

We next discuss approaches based on acknowledgment mechanisms, which primarily involve analyzing patterns and frequencies of acknowledgment (ACK) packets to identify packet reordering.

TCP employs a mechanism called Duplicate ACK (dup ACK), where it sends repeated acknowledgments to the corresponding sender for each expected received data packet that has not arrived. When the number of dup ACKs exceeds a certain threshold at the sender, various congestion control measures are triggered, resulting in a reduction in the congestion window (cwnd) size. However, cumulative acknowledgments are based on the entire system rather than a specific path, and it is challenging to differentiate whether the receipt of duplicate packets indicates their discard or reordering. Consequently, congestion control measures are often erroneously activated, leading to an unnecessary reduction in cwnd and the underutilization of available network resources. If not checked at the transport layer, these errors accumulate and propagate, resulting in a phenomenon known as "information dissipation". Therefore, it is necessary not only to differentiate packet discard from reordering but also to distinguish packet reordering between different paths. This problem requires modifications to TCP, leading to TCP variants. This section primarily introduces these techniques, categorizing TCP variants according to the strategies shown in Figure 8, and specific solutions are presented in Table 2. A ✓in the Table 2 indicates that the TCP variant in this row satisfies the corresponding column property in the table. Some results are taken from [69]. The authors of [70] contend that the susceptibility of TCP to packet reordering significantly hampers packet-level load balancing research. While various TCP variants have been developed to address this reordering issue, they have not gained widespread deployment or acceptance due to the absence of end-to-end transparency. As a response, the authors propose ORTA (Out-of-Order Robustness for TCP with Transparent Acknowledgment Intervention), a transparent and lightweight algorithm designed specifically to manage out-of-order delivery.

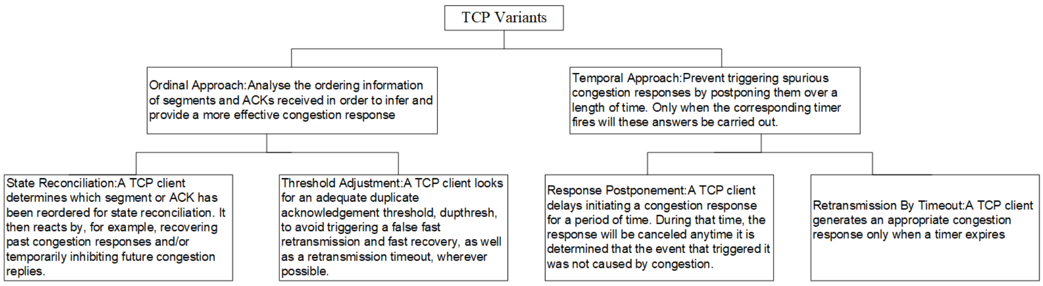

Figure 8. TCP variant classification.

Table 2. TCP variants.

Algorithms	Solution Strategy	Devices Involved			Additional Information Needed			Reduction in Spurious Retransmissions	Maintaining ACK-Clocking	Sustaining Larger Congestion Window	Fairer Estimation of RTT/RTO
		Source	Destination	Router	D-SACK	Reordered Bit	Timestamp /Sequence Number [71]				
Blanton–Allman Algorithms [36]	Threshold Adjustment	✓			✓			✓	✓	✓	
	Response Postponement	✓			✓			✓	✓	✓	
DSACK TCP [72]	State Reconciliation	✓			✓						
Eifel Algorithm [73]	State Reconciliation	✓					✓				
Lee–Park–Choi Algorithms [74]	Response Postponement	✓	✓					✓		✓	
Leung–Ma Algorithm [75]	Threshold Adjustment	✓			✓			✓	✓	✓	
Paxson Algorithm [27]	Response Postponement		✓					✓		✓	
RN-TCP [76]	Threshold Adjustment	✓	✓	✓		✓		✓		✓	
RR-TCP [43]	Threshold Adjustment	✓			✓			✓	✓	✓	✓
TCP-DCR [77]	Response Postponement	✓						✓	✓	✓	
TCP-DOOR [78]	State Reconciliation	✓	✓				✓	✓			
TCP-PR [15]	Retransmission by Timeout	✓						✓	✓	✓	✓
F-RTO [79]	Retransmission by Timeout	✓						✓			✓
RD-TCP [80]	State Reconciliation	✓		✓		✓		✓	✓	✓	
TCP-NCL [81]	State Reconciliation	✓						✓		✓	✓
	Retransmission by Timeout	✓						✓		✓	✓

UDP itself does not have congestion control and retransmission mechanisms, which makes UDP transmission unreliable. Therefore, improving its reliability has always been a research focus and hot topic in the field. One class of solutions aims to enhance UDP reliability by establishing simple acknowledgment, retransmission, and forward error correction mechanisms, resembling TCP mechanisms. These solutions are referred to as Enhanced Reliable UDP, these methods address packet reordering or out-of-order issues in different ways:

- SRUDP. SRUDP introduces acknowledgment, retransmission, and sequence alignment mechanisms. By using forward and backward sequence numbers in the protocol header, it ensures that packets are transmitted and acknowledged in the correct order, reducing the impact of packet reordering.
- RUDP. RUDP utilizes a request–response mechanism along with enhanced data service quality mechanisms such as improved congestion control and retransmission. These mechanisms help maintain the correct order of packets and ensure reliable transmission in the presence of packet loss and network congestion.
- KCP. KCP implements the Selective Repeat Automatic Repeat Request (ARQ) mechanism and offers features such as fast retransmission, delayed acknowledgment, and packet loss concession. It provides reliable byte stream transmission and utilizes forward error correction (FEC) using Reed–Solomon erasure codes to reduce the need for retransmissions, thereby minimizing data transmission delays caused by packet reordering.
- UDT. UDT builds upon UDP and implements TCP-like protocols and algorithms. It includes adjustments to TCP's congestion control algorithm and incorporates features such as Negative-ACK (NAK), ACK to ACK (ACK2), and logarithmic-based dynamic AIMD to handle packet reordering and congestion control.
- SCTP. SCTP is designed as a transport layer protocol that supports reliable transmission and message-oriented communication. It offers ordered or unordered message delivery and utilizes multiple network transmission paths. By avoiding TCP's SYN Flooding attack and utilizing multiple paths, SCTP reduces the impact of packet reordering in the network.
- QUIC. QUIC addresses packet reordering through flow control and packet loss recovery mechanisms using Packet Numbers. However, it introduces additional processing overhead and latency to handle out-of-order packets efficiently.
- uTP. uTP incorporates the LEDBAT congestion control algorithm, which detects network congestion based on latency. By detecting congestion early and making larger congestion avoidance adjustments, uTP minimizes the impact of packet reordering on user activities and ensures coexistence between background downloads and foreground operations.
- Enet. ENet provides a reliable ordered multichannel packet transmission mechanism, which helps maintain packet order and mitigate packet reordering issues.
- AWS SRD [82]. SRD leverages multiple network paths in modern data center networks to overcome load imbalance and inconsistent delays. While SRD does not preserve packet order, it sends packets through multiple paths, reducing the impact of packet reordering and avoiding path overload.
- HARP. HARP tracks the sending and receiving states of each packet using a self-developed packet numbering scheme. This allows for out-of-order reception and selective retransmission with low overhead, ensuring reliable transmission while handling packet reordering challenges.
- KUDP [83]. The Keyed User Datagram Protocol (KUDP) is designed for efficient data transmission. It boasts capabilities of precisely identifying lost packets and, notably, effectively reordering incoming non-conforming packets, thereby enhancing the efficiency of data flow.

In addition, the authors of [84] discuss options for the correction of packet reordering on the receiver side of RTP audiovisual streams transmitted on top of the unreliable UDP protocol. They also describe the design of a packet reordering correction unit suitable for efficient FPGA implementation and present the resource consumption of the proposed solution.

3.3.2. Method Based on Network Characteristics

To handle a large number of data flows, researchers have developed various efficient data structures based on the data plane. Tools such as HashPipe [85] and PRECISION [86] can handle reordering statistics of large data flows, but they cannot detect the reordering of individual packets within the flows. Some systems can detect TCP packet reordering on the data plane, such as Marple [87]. However, these methods require a per-flow state, which can consume more memory in practice. Algorithms proposed by Liu et al. [88] primarily focus on flows with a large number of disordered packets, while Zheng et al. [89] mitigate the lower bound on memory consumption by identifying a large number of reordering prefixes instead of flows.

3.3.3. Method Based on Machine Learning

This type of method attempts to identify packet reordering using machine learning algorithms. Fonseca et al. [90] proposed an optimal Bayesian packet loss detection method based on round-trip time and constructed an analytical performance model that incorporates general packet loss inference into TCP. The research showed that, for long-term traffic, based on measured round-trip time, high detection probability and low false alarm probability can generally be achieved. Using more general packet loss inference for realistic detection and false alarm probabilities, TCP throughput can be increased by up to 25%. Therefore, deep learning, support vector machines, and other algorithms can be used to predict packet reordering based on historical data. The advantages of this approach are its ability to adapt to network variations automatically and its high prediction accuracy. However, it requires a large amount of training data and demands significant computational resources.

3.4. Packet Reordering Tolerance

In many cases, network devices can only handle packet reordering issues by triggering congestion and retransmission mechanisms, which not only increase network latency but may also have a negative impact on the utilization efficiency of network bandwidth. However, if we deploy a sufficiently large buffer at the network terminal to temporarily store out-of-order packets, we can mitigate the adverse effects caused by packet reordering to some extent [24]. The basic idea of this approach is to perform packet reordering within a certain threshold, allowing the system to accept or tolerate a certain degree of disorder. Additionally, we can dynamically adjust the buffer size, introduce delay mechanisms for out-of-order packet acknowledgment, or adjust the number of required ACKs for retransmission to further reduce the occurrence of false retransmissions. When the network environment allows for a certain degree of packet loss or chooses to discard excessively delayed packets, the abnormal congestion window issue can also be effectively addressed.

For networks using the UDP transport protocol, different strategies can be employed to handle packet reordering and packet loss issues. First, the design goal of the UDP protocol is to prioritize low latency and low packet loss, often at the expense of data transmission quality. To address packet reordering and loss, we can approach the issue from both the UDP protocol itself and the application level. At the UDP protocol level, we can leverage buffer mechanisms to allow the application to recover from an out-of-order state as quickly as possible. However, it is important to recognize that in cases of severe reordering, it may not be feasible to allocate a buffer large enough to store all the packets. In such scenarios, trade-offs need to be made. From a temporal perspective, if the storage time of packets in the buffer exceeds a specified threshold, we can choose to discard subsequent out-of-order packets. From a spatial perspective, if the fixed-size buffer becomes full, we can also choose to discard subsequent out-of-order packets. However, such tolerance strategies for reordering and packet loss may not be conducive to maintaining system performance in cases of severe reordering.

4. Packet Reordering Metrics

4.1. Basic Metrics

Consider the two packet sequences in Table 3. It presents two sequences of packet reordering. In 'Sequence 1', packets 3 and 4 arrive after packet 5, thereby causing a reordering. Conversely, in 'Sequence 2', packet 3 arrives after packets 4 and 5.

Table 3. Two packet reordering sequences.

Sequence 1	1	2	5	3	4	6
Sequence 2	1	2	4	5	3	6

It is not difficult to see that both packet sequences have three out-of-order packets, resulting in a 50% reordering percentage. However, we can describe the situations of the two sequences from different perspectives:

- Describing based on the proportion of reordering: Packets 3, 4, and 5 are out of order in both sequences.
- Describing based on the increasing packet sequence numbers: In sequence 1, packets 3 and 4 are out of order, while in sequence 2, only packet 3 is out of order.
- Describing based on the absence of lower sequence numbers after higher ones: In sequence 1, only packet 5 is out of order, while in sequence 2, packets 4 and 5 are out of order.
- Describing based on the correspondence between packet sequence numbers and receiving indices: In both sequences, packets 3, 4, and 5 are out of order.

It is evident that describing the degree of reordering solely based on the percentage of out-of-order packets is ambiguous and does not provide a detailed and precise description of the depth of reordering. Therefore, it has certain limitations. To address the issue of packet reordering, it is crucial and necessary to choose appropriate metrics as tools and flexibly adjust them in different environments [91]. Packet reordering metrics are an important part of network performance analysis as they quantify the differences between the order of packet reception and the order of packet transmission in a network. The measurement of reordering typically involves calculating specific indicators and parameters, such as the percentage of reordered packets, the extent of reordering, or the delay caused by reordering. Packet reordering metrics not only focus on whether packets are out of order but also consider the degree of reordering and its potential impact. Therefore, they need to take into account multiple aspects of information, including the number of reordered packets, the degree of reordering, and the duration of reordering.

When overviewing and categorizing packet reordering metrics, we can classify them based on their evaluation objectives and methods. According to RFC 4737 [92], we can categorize them as shown in Table 4.

In the process of selecting and designing metrics, it is important to first clarify the basic characteristics of each metric. For example, we want a metric to capture the quantity and extent of reordering. This is the fundamental and most important characteristic of a metric. If a metric fails to meet this criterion, regardless of its other merits, it cannot be adopted. In this regard, the "Reordered Packet Ratio" is a good example as it captures the quantity of reordering while being less affected by packet loss and duplicate packets.

Furthermore, we expect metrics to be less sensitive to packet loss and duplicate packets. This is because reordered packets should represent out-of-order packets and should not include lost or duplicated packets. These scenarios should be measured by specialized metrics. Moreover, a deeper understanding of the advanced characteristics of each metric is necessary. This may include sensitivity to different network conditions, usefulness in specific application scenarios, and the computational complexity required to compute the metric. For example, the advanced characteristics of "Reordering Byte Offset" may include high sensitivity to reordering in large data streams and high usefulness in cloud computing and big data transfer scenarios.

Table 4. Basic metrics for packet reordering.

Metric	Category	Main Focus Usage	Scenario
A Reordered Packet Singleton Metric	Single packet reordering metric	Reordering of individual packets	Evaluating the degree of reordering for specific packets
Reordered Packet Ratio	Overall network performance reordering metric	Number of reordered packets over a period of time	Evaluating the overall reordering situation of the network
Reordering-free Runs	Overall network performance reordering metric	Number of consecutive packets without reordering	Measuring the stability of network performance
Reordering Late Time Offset	Reordering latency and offset metric	Packet delay caused by reordering	Measuring the impact of reordering on packet delay
Reordering Byte Offset	Measuring the impact of reordering on packet delay	Data offset caused by reordering	Measuring the impact of reordering on data offset
Reordering Extent	Reordering extent metric	Maximum deviation between sending and receiving order in a single reordering event	Evaluating the impact of a single reordering event
Metrics Focused on Receiver Assessment: A TCP-Relevant Metric	A TCP-Relevant Metric Protocol-specific reordering metric	A TCP-Relevant Metric Protocol-specific reordering metric	Evaluating the impact of packet reordering on network performance under the TCP protocol
Gaps between multiple Reordering Discontinuities	Gaps between multiple reordering discontinuities	The size of the gap between different reordering events in a large data flow	Evaluating changes in network conditions

Based on the understanding of each metric, a comprehensive evaluation needs to be conducted. This may involve assessing its efficiency in solving specific problems, adaptability to specific network environments, and usability in practical applications. For example, "Reordering-free Runs" may be highly effective in a stable network environment, but its efficiency may significantly decrease in an unstable network environment. Finally, it is necessary to compare different metrics, understand their strengths and weaknesses, and determine the situations in which they are most applicable. For example, "Reordering Late Time Offset" and "Reordering Byte Offset" both focus on the delay and offset caused by reordering, but the former emphasizes time delay, while the latter emphasizes data offset, making them suitable for different scenarios.

4.2. Advanced Metrics

Choosing and designing metrics is a multifaceted process that necessitates comprehensive assessment of their essential and advanced features, thorough evaluations, and comparative analyses. It is crucial to understand that no single metric can satisfy all application scenarios, and metrics should provide insight into the user experience or performance. While basic metrics, due to their simplicity and low computational overhead, are beneficial, they may fall short in addressing more complex requirements.

In response to these needs, this study synthesizes and consolidates representative advanced metrics, taking into account the aforementioned metric characteristics. Metrics such as Reordering Density (RD) have demonstrated effective performance and have been validated in several studies [35,93,94]. Other metrics, such as Reordering Buffer Density (RBD), have found applications in a variety of contexts [95–101]. Figure 9 summarizes these advanced metrics, with symbols indicating satisfaction level.

In an analysis of Figure 9, RD emerges as superior in performance, encompassing all basic and advanced characteristics, and outperforming other evaluation metrics. Its functional form enables real-time assessment of network reordering performance. However, traditional criteria for single-path transmission scenarios may be inadequate for capturing the reordering phenomena in multipath transmissions. Therefore, it becomes necessary to combine RD with a more concise and appropriate criterion for continuous monitoring of reordering.

Metrics /Properties	Reordering Measurement Capability	Data Loss Resilience	Usefulness/Information provided			Space Complexity	Time Complexity	Simplicity	Easy to Compute	Robustness	Broad Applicability
			TCP flow control	Resource allocation	Finding the root cause of failure						
Percentage of Late Packets	√/×	√/×	×	×	×	O(N)	O(N)/O(N²)	×	√	×	×
Reorder Density	√	√	√	√	√	Constant	O(N)	√	√	√	√
Reorder Entropy	√	√	√	√	√	Constant	O(N)	√	√	√	√
Reorder Buffer-occupancy Density	√/×	√/×	×	×	×	Constant	O(N)	√	√	×	×
Reordering Extent	√/×	√/×	√	×	×	O(N)	O(N²)	√	√	×	×
n-Reordering	√/×	√/×	×	×	√	O(N)	O(N²)	√/×	×	×	×
Mean Displacement of Packets	√/×	√/×	×	×	√	O(N)	O(N)	√	√	√	√

Figure 9. Advanced metrics summary. The '✓' symbol represents that the corresponding metric possesses the indicated property, while the '×' symbol indicates that it does not. The symbol between '✓' and '×' indicates that it partially satisfies the property.

The importance of robust metrics in network analysis cannot be overstated. While traditional metrics such as the Percentage of Late Packets (PL), Mean Displacement of Packets (MD), and Mean Displacement of Late Packets (ML) offer valuable insights, the complexities of packet reordering in 6G networks demand more sophisticated solutions. Two such emerging metrics of interest are Reordering Entropy (RE) [102] and Reorder Density (RD). Reordering Entropy applies the concept of entropy to measure the randomness or disorderliness in packet reordering. With its high informational content, practicality, lower computational cost, and encompassing characteristics, RE serves as an effective tool to gauge the degree of chaos, reordering characteristics, and trends in the packet sequence. Before detailing these two metrics, we give a few definitions of the terms.

- Expected Packet (E): This refers to the sequence number of the next expected packet. If E is the maximum number, then all packets with a sequence number less than E should have already arrived or been confirmed as lost. Packets arriving with a sequence number higher than the current expected packet will be buffered.
- Receive Index (RI): The Receive Index RI $(1, 2, \ldots)$ is allocated in the order of arrival at the destination. It is not assigned to duplicate packets and skips lost packets. In the absence of disorder, the sequence number and the Receive Index for each packet are the same.
- Displacement (D): This is the difference between the sequence number and RI, calculated as $RI[i] - i$.
- Displacement Threshold (D_T): Any packet exceeding D_T is considered lost or duplicate. In theory, to track a duplicate packet, all arriving and lost packets must be tracked. However, practically speaking, it is sufficient to consider a window of sequence numbers for lost packets. If D_T is too large, it increases memory size; if it is too small, reordered packets might also be considered lost. Thus, D_T should be specified according to the sequence number and length, determining when a packet is considered lost or redundant.
- Displacement Frequency (FD): This refers to the displacement of the number k of arriving packets, where $k \in [-D_T, D_T]$.

RD is defined as the distribution of Displacement Frequency $FD[k]$, i.e., the delay and advancement relative to the original position [103]. If the receive index assigned to packet m is $(m + d_m)$ and $d_m \neq 0$, a "reordering" event has occurred, represented by $r(m, d_m)$. If this offset $d_m > 0$, packet m is delayed; if $d_m < 0$, packet m is early; if $d_m = 0$, packet m is in order. Therefore, packet reordering in the packet sequence is entirely represented by the union of reordering events $S[k]$, termed as the "reordering set".

Let $S[k]$ represent the set of reordering events with displacement equal to k, i.e., $S[k] = r(m, d_m)|$ This is normalized relative to N', where N' is the length of the received sequence, ignoring lost and duplicate packets. N' equals the sum of k in $[-D_T, D_T]$, which is $(FD[k])$. Therefore, RD is also defined as a histogram of d_m values, normalized according to the total number of packets.

- When $k \neq 0$, $S[k] = \cup_m (r(m, d_m)|d_m \neq 0)$, and $F[i]$ is the number of arrival instances where buffer i is occupied, which is

$$RD[k] = |S[k]|/N' = \frac{F[i]}{\sum_j F[j]}. \quad (1)$$

- When $k = 0$, $RD[0]$ corresponds to packets where the receive index is identical to the sequence number, which is

$$RD[0] = 1 - \sum_{k \neq 0} |S[k]|/N'. \quad (2)$$

The concept of entropy originated in the field of physics, representing "changes in the intrinsic properties of a system". Shannon extended the notion of entropy from theoretical physics to information theory, where it is often referred to as "information entropy" or "Shannon entropy". This concept can be used to define randomness or disorder. The expected information content of a probability distribution is calculated by weighting information values according to their respective probabilities p_i:

$$E = \sum_{n}^{i=1} p_i log_e * (1/p_i). \quad (3)$$

In the measurement of packet reordering, the concept of entropy can reflect the level of disorder, the characteristics of reordering, and the trends within a time interval. When combined with the most mature existing metric, RD, we present the calculation formula for RE based on RD:

$$RE = -\sum_{i=+DT}^{i=-DT} (RD[i] * lnRD[i]) \quad (4)$$

This concept effectively assesses the complexity and disorderliness of packet reordering, providing a quantifiable measure of packet sequence chaos, reordering characteristics, and temporal trends. When combined with the mature metric RD, it provides a powerful tool for comprehensively monitoring packet reordering. The calculation for RE allows for the application of this theoretical concept to practical network metrics, contributing to the ongoing development of comprehensive and sophisticated network monitoring tools. When RE is paired with RD, it can facilitate continuous and comprehensive monitoring of reordering, showing promise for trend monitoring. Reorder Density, on the other hand, allows for real-time assessment of network reordering performance. It incorporates all basic and advanced characteristics related to packet reordering, outperforming other evaluation metrics. However, for complex multipath transmission scenarios, RD may need to be combined with other concise and appropriate criteria to provide a more holistic view of reordering.

These advanced metrics are instrumental not only in describing the packet reordering phenomenon but also in capturing characteristic trends and dynamics. This improved understanding fosters efficient and stable network operations, crucial for the successful implementation of 6G networks. Moreover, as we navigate through the unique network environments and application scenarios within the 6G landscape, deepening our understanding of the mechanisms and factors influencing packet reordering is paramount. This understanding opens up exciting opportunities for future research, such as the development of new metrics and management strategies [104]. Advancements in AI and machine learning technologies hold promise for amplifying the intelligence and automation of packet reordering management. Potential applications include predictive and adaptive control of packet reordering through machine learning models. Therefore, it is clear that the study of packet reordering remains an active and essential area of research [89].

5. Challenges and Application Prospects of Packet Reordering in 6G Vehicular Networks

Vehicular networks within the 6G environment are anticipated to deliver ultra-high data rates (up to 100 Gbps), ultra-low latency (sub-millisecond), and massive connectivity. However, these advancements also intensify the challenges associated with packet reordering.

- Ultra-High Data Rates and Packet Reordering. The tremendous data rates in 6G vehicular networks suggest an incredible boost in packet transmission speeds. Consequently, this may amplify the packet reordering problem, and even minor network fluctuations could cause disparities between the sequence of packet reception and transmission in high-speed networks.
- Massive Connectivity and Packet Reordering. Vehicular networks in the 6G era are expected to accommodate a vast number of simultaneous connections, which brings complexity to packet reordering. With numerous devices transmitting data concurrently, congestion and flow control issues can become more pronounced, thus heightening the chances of packet reordering. Furthermore, with the presence of many devices, packet reordering identification and rectification become more challenging. For instance, [13] discovered that existing packet reordering detection algorithms perform inadequately in network environments with a high number of participating devices.

In such environments, it is crucial to deeply explore packet reordering challenges and consider them thoroughly in the design and optimization of networks.

- User-Centric Network Design. Within 6G vehicular networks, user-centric network design will be a significant trend, taking into account the user experience requirements in various contexts and scenarios. Applications such as in-car VR/AR, high-definition video streaming, and smart car interiors can be considerably affected by packet reordering, making it a priority to address in network design and optimization. Effective prevention and resolution approaches should be proposed.
- Increasing Demands for High Bandwidth and Low Latency. The widespread adoption of 6G networks will further amplify the demands for high bandwidth and low latency, imposing greater challenges on packet reordering. This necessitates the evolution of more efficient reordering detection and repair techniques to meet these stringent network performance requirements.
- Network Automation and Intelligence. The development of 6G vehicular networks will advance network automation and intelligence, providing new opportunities for resolving packet reordering issues. Leveraging machine learning and artificial intelligence technologies, network systems can automatically detect and rectify packet reordering problems, thereby enhancing network performance and stability.
- Data-Driven Network Optimization. In the 6G network environment, data-driven network optimization will become increasingly important. By collecting and analyzing network data, we can gain more profound insights into the causes and impacts of packet reordering, leading to more effective solutions.
- Integration of Multiple Services. Within 6G vehicular networks, the integration of diverse network services such as IoT-based vehicular services, autonomous driving, and smart traffic management will be more comprehensive. Such integrations impose higher requirements on network stability and timely packet processing, which places greater emphasis on packet reordering issues.

Overall, the challenges and application prospects of packet reordering in 6G vehicular networks underline the importance of addressing issues related to ultra-high data rates, massive connectivity, user-centric design, high bandwidth and low latency demands, network automation and intelligence, data-driven optimization, and the integration of multiple services. Effectively confronting these challenges and exploiting the application prospects will be pivotal for the successful deployment and operation of packet reordering techniques in 6G vehicular networks.

6. Conclusions

This paper conducted an in-depth exploration into the current state of research, causes, impacts, and solutions of packet reordering. It underlined the pervasiveness of packet reordering in network transmission and its profound impact on network performance, with particular emphasis on its influence on real-time applications and those demanding high-quality data transmission, such as vehicular networks. Current strategies for handling packet reordering, including prediction, avoidance, identification, and tolerance, were scrutinized alongside their applications and limitations in intricate network environments. The paper also shed light on metrics for packet reordering, illuminating their role in network performance evaluation and optimization.

The swift advancement of network technologies, including the adoption of novel network architectures such as cloud computing, edge computing, the Internet of Things (IoT), and intelligent devices in vehicular networks, are likely to intensify packet reordering occurrences. While existing research has addressed packet reordering to a certain extent, several challenges linger. Notably, most of the current research primarily concentrates on traditional TCP end-to-end packet reordering measurement and analysis, offering limited insights into new or heterogeneous network environments, such as vehicular networks. This signals a need for more extensive research and in-depth validation.

In the context of vehicular networks within 6G, the paper discussed the challenges and trends, identifying the impact of features such as ultra-high data rates and massive connectivity on packet reordering. Moreover, it outlined potential development trends, including user-centric network design, escalating demands for high bandwidth and low latency, network automation and intelligence, data-driven network optimization, and integration of various vehicular services. These considerations highlight the importance of and need for continued research and development in packet reordering.

7. Future Directions and Limitations

To tackle the challenges discussed, future research in the realm of packet reordering, particularly in vehicular networks, can concentrate on the following areas:

- Deep exploration of emerging network environments. An in-depth investigation into novel network architectures such as cloud computing, edge computing, and the Internet of Things (IoT), along with heterogeneous network environments, is essential. These network landscapes, especially vehicular networks, introduce unique hurdles concerning packet reordering, requiring customized solutions.
- Improvement of feasibility and efficiency of deploying novel strategies. Future studies should strive to unify theoretical models with practical applications in vehicular networks. Investigating the practical affect of theoretical models, assessing the feasibility and efficiency of novel strategies and technologies, and addressing the challenges arising from their deployment in real-world vehicular networks are crucial steps.
- Studies on large-scale and complex vehicular network environments. The verification and optimization of solutions for packet reordering become increasingly challenging in large-scale and complicated vehicular network settings. Further research is needed to effectively manage packet reordering in such environments and develop scalable and efficient solutions.

While this paper aims for impartiality, it is worth acknowledging that it is based on our interpretation and synthesis of the existing literature, which could introduce some subjectivity. Furthermore, considering the majority of referenced works were published over five years ago, there are inherent limitations regarding the temporal scope and source materials. Consequently, readers should exercise discretion when interpreting the findings and extrapolating them to the context of contemporary vehicular networks.

Author Contributions: Conceptualisation: J.L., X.G., Y.O. and T.F.; methodology: X.Z. and P.K.; validation: J.L. and X.Z.; investigation: J.L., X.Z. and P.K.; resources: Y.O., X.G. and T.F.; data curation: J.L. and Y.Z.; writing—original draft preparation: J.L.; writing—review and editing: J.L., X.Z., X.G., P.K., Y.Z., Y.O. and T.F.; visualisation: J.L. and Y.Z.; supervision: X.G. and T.F.; project administration: X.G. and T.F.; funding acquisition: X.G. and T.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ge, X.; Cheng, H.; Guizani, M.; Han, T. 5G wireless backhaul networks: Challenges and research advances. *IEEE Netw.* **2014**, *28*, 6–11. [CrossRef]
2. Yang, P.; Xiao, Y.; Xiao, M.; Li, S. 6G Wireless Communications: Vision and Potential Techniques. *IEEE Netw.* **2019**, *33*, 70–75. [CrossRef]
3. David, K.; Berndt, H. 6G Vision and Requirements: Is There Any Need for Beyond 5G? *IEEE Veh. Technol. Mag.* **2018**, *13*, 72–80. [CrossRef]
4. Saad, W.; Bennis, M.; Chen, M. A vision of 6G wireless systems: Applications, trends, technologies, and open research problems. *IEEE Netw.* **2019**, *34*, 134–142. [CrossRef]
5. Ghasemirahni, H.; Barbette, T.; Katsikas, G.P.; Farshin, A.; Roozbeh, A.; Girondi, M.; Chiesa, M.; Maguire, G.Q., Jr.; Kostić, D. Packet order matters! improving application performance by deliberately delaying packets. In Proceedings of the 19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22), Renton, WA, USA, 4–6 April 2022 ; pp. 807–827.
6. Yang, W.; Cai, L.; Shu, S.; Pan, J. Scheduler Design for Mobility-aware Multipath QUIC. In Proceedings of the GLOBECOM 2022—2022 IEEE Global Communications Conference, Rio de Janeiro, Brazil, 4–8 December 2022 ; pp. 2849–2854.
7. Hayes, D.A.; Ros, D.; Ytrehus, O. Proxy Path Scheduling and Erasure Reconstruction for Low Delay mmWave Communication. *IEEE Commun. Lett.* **2023**, *27*, 1649–1653. [CrossRef]
8. Jayasumana, A.; Piratla, N.; Labs, D.T.; Banka, T.; Bare, A.; Whitner, R. Improved Packet Reordering Metrics. RFC 5236, 1–26. Available online: https://www.rfc-editor.org/info/rfc5236 (accessed on 13 June 2023). [CrossRef]
9. Leung, K.C.; Lai, C.; Li, V.; Yang, D. A packet-reordering solution to wireless losses in transmission control protocol. *Wirel. Netw.* **2013**, *19*, 1577–1593. [CrossRef]
10. Yi, W.; Lu, G.; Xing, L. *A Study of Internet Packet Reordering*; Springer: Berlin/Heidelberg, Germany, 2004.
11. Paxson, V.; Almes, G.; Mahdavi, J.; Mathis, M. Framework for IP Performance Metrics. RFC 2330: 1–40. 1998. Available online: https://www.rfc-editor.org/info/rfc2330 (accessed on 13 June 2023). [CrossRef]
12. Khayam, S.A.; Karande, S.; Radha, H.; Loguinov, D. Performance analysis and modeling of errors and losses over 802.11b LANs for high-bit-rate real-time multimedia. *Signal Process Image Commun.* **2003**, *18*, 575–595. [CrossRef]
13. Gharai, L.; Perkins, C.; Lehman, T. Packet reordering, high speed networks and transport protocol performance. In Proceedings of the Proceedings. 13th International Conference on Computer Communications and Networks (IEEE Cat. No.04EX969), Chicago, IL, USA, 11–13 October 2004; pp. 73–78.
14. Bennett, J.C.R.; Partridge, C. Packet reordering is not pathological network behavior. *IEEE/ACM Trans. Netw.* **1999**, *7*, 789–798. [CrossRef]
15. Bohacek, S.; Hespanha, J.P.; Lee, J.; Lim, C.; Obraczka, K. A new TCP for persistent packet reordering. *IEEE/ACM Trans. Netw.* **2006**, *14*, 369–382. [CrossRef]
16. Pham, P.P.; Perreau, S. Performance analysis of reactive shortest path and multipath routing mechanism with load balance. In Proceedings of the INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications, IEEE Societies, San Franciso, CA, USA, 30 March–3 April 2003; pp. 251–259.
17. Chao, W.; Zhang, X.; Chen, W.; Niu, X. Load Balancing Algorithm Using Flow Chopping to Avoid Packet Reordering. In Proceedings of the 2009 International Forum on Information Technology and Applications, Chengdu, China, 15–17 May 2009; pp. 193–197.
18. Tinta, S.; Mohr, A.; Wong, J. Characterizing end-to-end packet reordering with UDP traffic. In Proceedings of the 14th IEEE Symposium on Computers and Communications (ISCC 2009), Sousse, Tunisia, 5–8 July 2009; pp. 321–324.
19. Jie, F.; Ouyang, Z.; Xu, L.; Ramamurthy, B. Packet reordering in high-speed networks and its impact on high-speed TCP variants. *Comput. Commun.* **2009**, *32*, 62–68.
20. Rottenstreich, O.; Li, P.; Horev, I.; Keslassy, I.; Kalyanaraman, S. The Switch Reordering Contagion: Preventing a Few Late Packets from Ruining the Whole Party. *IEEE Trans. Comput.* **2014**, *63*, 1262–1276. [CrossRef]

21. Lelarge, M. Packet reordering in networks with heavy-tailed delays. *Math. Methods Oper. Res.* **2008**, *67*, 341–371. [CrossRef]
22. Bellardo, J.M.; Savage, S. Measuring Packet Reordering. In Proceedings of the 2nd ACM SIGCOMM Internet Measurement Workshop, IMW 2002, Marseille, France, 6–8 November 2002; pp. 97–105.
23. Nebat, Y.; Sidi, M. Analysis of resequencing in downloads. *Int. J. Commun. Syst.* **2010**, *16*, 735–757. [CrossRef]
24. Kaspar, D.; Evensen, K.; Hansen, A.F.; Engelstad, P.; Halvorsen, P.; Griwodz, C. An analysis of the heterogeneity and IP packet reordering over multiple wireless networks. In Proceedings of the 14th IEEE Symposium on Computers and Communications (ISCC 2009), Sousse, Tunisia, 5–8 July 2009; pp. 637–642.
25. Hu, F.; Sharma, N.K. Enhancing wireless internet performance. *IEEE Commun. Surv. Tutor.* **2009**, *4*, 2–15. [CrossRef]
26. Arthur, C.M.; Girma, D.; Harle, D.; Lehane, A. The effects of packet reordering in a wireless multimedia environment. In Proceedings of the 1st IEEE International Symposium on Wireless Communication Systems, ISWCS 2004, Mauritius, 20–22 September 2004; pp. 453–457.
27. Paxson, V. End-to-end Internet packet dynamics. *IEEE/ACM Trans. Netw.* **1999**, *7*, 277–292. [CrossRef]
28. Yang, S.; Lin, B.; Tune, P.; Xu, J.J. A simple re-sequencing load-balanced switch based on analytical packet reordering bounds. In 2017 IEEE Conference on Computer Communications, INFOCOM 2017, Atlanta, GA, USA, 1–4 May 2017; pp. 1–9.
29. Li, F.; Wang, X.; Pan, T.; Yang, J. A Case Study of IPv6 Network Performance: Packet Delay, Loss, and Reordering. *Math. Probl. Eng.* **2017**, *2017*, 1–10. [CrossRef]
30. Gao, C.; Ling, Z.; Yuan, Y. Packet reordering analysis for concurrent multipath transfer. *Int. J. Commun. Syst.* **2014**, *27*, 4510–4526. [CrossRef]
31. Alheid, A.; Doufexi, A.; Kaleshi, D. Packet reordering response for MPTCP under wireless heterogeneous environment. In Proceedings of the 23rd International Conference on Telecommunications, ICT 2016, Thessaloniki, Greece, 16–18 May 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 1–6.
32. Hamza, B.J.; Kyun, N.C.; Noordin, N.K.; Rasid, M.F.A.; Ismail, A.; Tahir, Y.H. Enhancement of packet reordering in a mobile stream control transmission protocol for a heterogeneous wireless network vertical handover. *J. High Speed Netw.* **2010**, *17*, 163–173. [CrossRef]
33. Amend, M.; Moreno, N.R.; Pieskä, M.; Kassler, A.; Brunström, A.; Rakocevic, V.; Johnson, S. In-network Support for Packet Reordering for Multiaccess Transport Layer Tunneling. In Proceedings of the 2022 IEEE 11th IFIP International Conference on Performance Evaluation and Modeling in Wireless and Wired Networks (PEMWN), Rome, Italy, 8–10 November 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–6.
34. Prabhavat, S.; Nishiyama, H.; Ansari, N.; Kato, N. On the performance analysis of traffic splitting on load imbalancing and packet reordering of bursty traffic. In Proceedings of the 2009 IEEE International Conference on Network Infrastructure and Digital Content, Beijing, China, 6–8 November 2009; pp. 236–240.
35. Piratla, N.M.; Jayasumana, A.P.; Bare, A.A.; Banka, T. Reorder buffer-occupancy density and its application for measurement and evaluation of packet reordering. *Comput. Commun.* **2007**, *30*, 1980–1993. [CrossRef]
36. Blanton, E.; Allman, M. On Making TCP More Robust to Packet Reordering. *ACM SIGCOMM Comput. Commun. Rev.* **2002**, *32*, 20–30. [CrossRef]
37. Postel, J. Transmission Control Protocol. *RFC* **1981**, *2*, 595–599. Available online: https://www.rfc-editor.org/info/rfc0793 (accessed on 11 June 2023). [CrossRef]
38. Kasimsetti, S.P.; Hussain, A. Enhanced packet reordering procedure to improve TCP communication. *Int. J. Pervasive Comput. Commun.* **2022**, *18*, 98–113. [CrossRef]
39. Allman, M. TCP Congestion Control. RFC 2009. Available online: https://www.rfc-editor.org/info/rfc5681 (accessed on 13 June 2023). [CrossRef]
40. Paxson, V.; Allman, M.J. *RFC 2988—Computing TCP's Retransmission Timer*; RFC:2012. Available online: https://www.rfc-editor.org/info/rfc2988 (accessed on 11 June 2023). [CrossRef]
41. Sarolahti, P. Forward RTO-Recovery (F-RTO): An Algorithm for Detecting Spurious Retransmission Timeouts with TCP and the Stream Control Transmission Protocol (SCTP). *RFC4138*. Available online: https://www.rfc-editor.org/info/rfc4138 (accessed on 11 June 2023). [CrossRef]
42. Floyd, S.; Fall, K. Promoting the Use of End-to-End Congestion Control in the Internet. *IEEE/ACM Trans. Netw.* **1999**, *7*, 458–472. [CrossRef]
43. Ming, Z.; Karp, B.; Floyd, S.; Peterson, L.L. RR-TCP: A Reordering-Robust TCP with DSACK. In Proceedings of the 11th IEEE International Conference on Network Protocols (ICNP 2003), Atlanta, GA, USA, 4–7 November 2003; pp. 95–106.
44. Jacobson, V. Congestion avoidance and control. *ACM SIGCOMM Comput. Commun. Rev.* **1988**, *18*, 314–329. [CrossRef]
45. Laor, M.; Gendel, L. The effect of packet reordering in a backbone link on application throughput. *IEEE Netw.* **2002**, *16*, 28–36. [CrossRef]
46. Dharmapurikar, S.; Paxson, V. Robust TCP stream reassembly in the presence of adversaries. In Proceedings of the 14th USENIX Security Symposium, Baltimore, MD, USA, 31 July–5 August 2005.
47. Varghese, G.; Fingerhut, J.A.; Bonomi, F. Detecting evasion attacks at high speeds without reassembly. In Proceedings of the ACM SIGCOMM 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, Pisa, Italy, 11–15 September 2006; p. 327.

48. Vutukuru, M.; Balakrishnan, H.; Paxson, V. Efficient and Robust TCP Stream Normalization. In Proceedings of the Security and Privacy (SP 2008), Oakland, CA, USA, 18–21 May 2008; IEEE: Piscataway, NJ, USA, 2008.
49. Zhao, F.; Shin, Y.; Wu, S.F.; Johnson, H.; Nilsson, A. RBWA: An efficient random-bit window-based authentication protocol. In Proceedings of the Global Telecommunications Conference, 2003. GLOBECOM '03, San Francisco, CA, USA, 1–5 December 2003; pp. 1379–1383.
50. El-Atawy, A.; Al-Shaer, E. Building covert channels over the packet reordering phenomenon. In *Proceedings of the IEEE INFOCOM 2009*; IEEE: Piscataway, NJ, USA, 2009; pp. 2186–2194.
51. Wu, J.; Bo, C.; Shang, Y.; Huang, J.; Chen, J. A novel scheduling approach to concurrent multipath transmission of high definition video in overlay networks. *J. Netw. Comput. Appl.* **2014**, *44*, 17–29. [CrossRef]
52. Chebrolu, K.; Rao, R.R. Bandwidth aggregation for real-time applications in heterogeneous wireless networks. *IEEE Trans. Mob. Comput.* **2006**, *5*, 388–403. [CrossRef]
53. Jurca, D.; Frossard, P. Video Packet Selection and Scheduling for Multipath Streaming. *IEEE Trans. Multimed.* **2007**, *9*, 629–641. [CrossRef]
54. Korzun, D.; Kuptsov, D.; Gurtov, A. A Simulation Study of the Stochastic Compensation Effect for Packet Reordering in Multipath Data Streaming. In Proceedings of the 2015 IEEE European Modelling Symposium, EMS 2015, Madrid, Spain, 6–8 October 2015; pp. 409–414.
55. Wu, D.; Wang, R.; Zhen, Y. Link stability-aware reliable packet transmitting mechanism in mobile ad hoc network. *Int. J. Commun. Syst.* **2012**, *25*, 1568–1584. [CrossRef]
56. Sunitha, D.; Nagaraju, A.; Narsimha, G. Cross-layer based routing protocol and solution to packet reordering for TCP in MANET. *Clust. Comput.* **2019**, *22*, 10809–10816. [CrossRef]
57. Zhou, X.; Zhang, D.; Yang, Y.; Obaidat, M.S. Network-coded multiple-source cooperation aided relaying for free-space optical transmission. *Int. J. Commun. Syst.* **2012**, *25*, 1465–1478. [CrossRef]
58. Ha, N.V.; Tsuru, M. TCP with Network Coding Performance Under Packet Reordering. In Proceedings of the Advances in Internet, Data and Web Technologies, 7th International Conference on Emerging Internet, Data and Web Technologies (EIDWT-2019), Fujairah Campus, United Arab Emirates, 26–28 February 2019 ; Springer: Berlin/Heidelberg, Germany, 2019; Volume 29, pp. 552–563.
59. Hu, J.; Ruan, C.; Wang, L.; Alfarraj, O.; Tolba, A. Coding-Based Distributed Congestion-Aware Packet Spraying to Avoid Reordering in Data Center Networks. *IEEE Access* **2021**, *9*, 35539–35548. [CrossRef]
60. Huang, J.; Lyu, W.; Li, W.; Wang, J.; He, T. Mitigating Packet Reordering for Random Packet Spraying in Data Center Networks. *IEEE/ACM Trans. Netw.* **2021**, *29*, 1183–1196. [CrossRef]
61. Mohammadpour, E.; Boudec, J.L. On Packet Reordering in Time-Sensitive Networks. *IEEE/ACM Trans. Netw.* **2022**, *30*, 1045–1057. [CrossRef]
62. Iqbal, M.F.; Holt, J.; Ryoo, J.H.; John, L.K.; Veciance, G.D. Flow Migration on Multicore Network Processors: Load Balancing While Minimizing Packet Reordering. In Proceedings of the 42nd International Conference on Parallel Processing, ICPP 2013, Lyon, France, 1–4 October 2013; pp. 150–159.
63. Prabhavat, S.; Nishiyama, H.; Ansari, N.; Kato, N. Effective Delay-Controlled Load Distribution over Multipath Networks. *IEEE Trans. Parallel Distrib. Syst.* **2011**, *22*, 1730–1741. [CrossRef]
64. Elwalid, A.; Jin, C.; Low, S.; Widjaja, I. MATE: MPLS adaptive traffic engineering. In Proceedings of the IEEE INFOCOM 2001, Conference on Computer Communications, Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, Twenty Years into the Communications Odyssey, Anchorage, AK, USA, 22–26 April 2001; pp. 1300–1309.
65. You-Jun, B.U. Load Balancing Algorithm Using Flow Splitting to Avoid Packet Reordering. *Comput. Sci.* **2010**, *37*, 67–70.
66. Kandula, S.; Katabi, D.; Davie, B.S.; Charny, A. Walking the Tightrope: Responsive Yet Stable Traffic Engineering. In Proceedings of the ACM SIGCOMM 2005 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, Philadelphia, PA, USA, 22–26 August 2005; pp. 253–264.
67. Hao, W.; Xie, H.; Qiu, L.; Yang, R.Y.; Greenberg, A.G. COPE: Traffic engineering in dynamic networks. In Proceedings of the ACM SIGCOMM 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, Pisa, Italy, 11–15 September 2006; pp. 99–110.
68. Roughan, M. Public review for dynamic load balancing without packet reordering. *ACM SIGCOMM Comput. Commun. Rev.* **2007**, *37*, 51.
69. Leung, K.C.; Li, V.O.K.; Yang, D. An Overview of Packet Reordering in Transmission Control Protocol (TCP): Problems, Solutions, and Challenges. *IEEE Trans. Parallel Distrib. Syst.* **2007**, *18*, 522–535. [CrossRef]
70. Ayar, T.; Budzisz, L.; Rathke, B. A Transparent Reordering Robust TCP Proxy To Allow Per-Packet Load Balancing in Core Networks. In Proceedings of the 2018 9th International Conference on the Network of the Future (NOF), Poznan, Poland, 19–21 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 17–24.
71. Jacobson, V.; Braden, R.; Borman, D. TCP Extensions for High Performance. RFC 1997. Available online: https://www.rfc-editor.org/info/rfc1323 (accessed on 11 June 2023). [CrossRef]
72. Podolsky, M. An Extension to the Selective Acknowledgement (SACK) Option for TCP. IETF RFC 2000. Available online: https://www.rfc-editor.org/info/rfc2883 (accessed on 11 June 2023). [CrossRef]

73. Katz, R. The Eifel Algorithm: Making TCP Robust Against Spurious Retransmissions. *ACM SIGCOMM Comput. Commun. Rev.* **2000**, *30*, 30–36.
74. Lee, Y.; Park, I.; Choi, Y. Improving TCP Performance in Multipath Packet Forwarding Networks. *J. Commun. Netw.* **2013**, *4*, 148–157. [CrossRef]
75. Leung.; Ka-Cheong.; Changming. Enhancing TCP performance to persistent packet reordering. *Commun. Netw. J.* **2005**, *7*, 385–393. [CrossRef]
76. Sathiaseelan, A.; Radzik, T. *Improving the Performance of TCP in the Case of Packet Reordering*; Springer: Berlin/Heidelberg, Germany, 2004.
77. Bhandarkar, S.; Reddy, A. TCP-DCR: Making TCP robust to non-congestion events. In *NETWORKING 2004: Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communication, Proceedings of the Third International IFIP-TC6 Networking Conference, Athens, Greece, 9–14 May 2004*; Springer: Berlin/Heidelberg, Germany, 2004.
78. Feng, W.; Zhang, Y. Improving TCP performance over mobile ad hoc networks with out-of-order detection and response. In Proceedings of the 3rd ACM Interational Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc 2002, Lausanne, Switzerland, 9–11 June 2002; pp. 217–225.
79. Kojo, M.; Yamamoto, K.; Hata, M.; Docomo, N. *Forward RTO-Recovery (F-RTO): An Algorithm for Detecting Spurious Retransmission Timeouts with TCP*; Social Science Electronic Publishing: Rochester, NJ, USA, 2005; Volume 16, pp. 29–35.
80. Sathiaseelan, A.; Radzik, T. RD-TCP: Reorder Detecting TCP. In *High-Speed Networks and Multimedia Communications, Proceedings of the 6th IEEE International Conference HSNMC 2003, Estoril, Portugal, 23–25 July 2003* ; Springer: Berlin/Heidelberg, Germany, 2003.
81. Lai, C.; Leung, K.; Li, V. TCP-NCL: A unified solution for TCP packet reordering and random loss. In Proceedings of the 2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications, Tokyo, Japan, 13–16 September 2009; IEEE: Piscataway, NJ, USA, 2009.
82. Shalev, L.; Ayoub, H.; Bshara, N.; Sabbag, E. A cloud-optimized transport protocol for elastic and scalable hpc. *IEEE Micro* **2020**, *40*, 67–73. [CrossRef]
83. Gil, F.M.; Garcia, N.M.; Matos, B.; Pombo, N.; Goleva, R.; Dobre, C. Identifying Packet Loss and Reordering Packets in Keyed UDP Transmissions. In Proceedings of the GLOBECOM (Workshops), Taipei, Taiwan, 8–10 December 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–5.
84. Benes, T.; Ubik, S.; Halák, J. Packet Reordering Correction for Low-Latency Network Applications. In Proceedings of the MECO, Budva, Montenegro, 7–10 June 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–5.
85. Sivaraman, V.; Narayana, S.; Rottenstreich, O.; Muthukrishnan, S.; Rexford, J. Heavy-hitter detection entirely in the data plane. In Proceedings of the Symposium on SDN Research, SOSR 2017, Santa Clara, CA, USA, 3–4 April 2017; pp. 164–176.
86. Metwally, A.; Agrawal, D.; El Abbadi, A. Efficient computation of frequent and top-k elements in data streams. In Proceedings of the *Database Theory-ICDT 2005, 10th International Conference, Edinburgh, UK, 5–7 January 2005*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 398–412.
87. Narayana, S.; Sivaraman, A.; Nathan, V.; Goyal, P.; Arun, V.; Alizadeh, M.; Jeyakumar, V.; Kim, C. Language-directed hardware design for network performance monitoring. In Proceedings of the Conference of the ACM Special Interest Group on Data Communication, SIGCOMM 2017, Los Angeles, CA, USA, 21–25 August 2017; pp. 85–98.
88. Liu, Z.; Zhou, S.; Rottenstreich, O.; Braverman, V.; Rexford, J. Memory-efficient performance monitoring on programmable switches with lean algorithms. In Proceedings of the 1st Symposium on Algorithmic Principles of Computer Systems, APOCS 2020, Salt Lake City, UT, USA, 8 January 2020; pp. 31–44.
89. Zheng, Y.; Yu, H.; Rexford, J. Detecting TCP Packet Reordering in the Data Plane. *arXiv* **2022**, arXiv:2301.00058.
90. Fonseca, N.; Crovella, M. Bayesian packet loss detection for TCP. In Proceedings of the INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies, Miami, FL, USA, 13–17 March 2005; Volume 3, pp. 1826–1837.
91. Torres, P.R., Jr.; Ribeiro, E.P. Packet Reordering Metrics to Enable Performance Comparison in IP-Networks. *J. Comput. Netw. Commun.* **2020**, *2020*, 8465191:1–8465191:8. [CrossRef]
92. Morton, A.; Ciavattone, L.; Ramachandran, G.; Shalunov, S.; Perser, J. Packet Reordering Metrics. *Technical Report*; 2006. Available online: https://www.rfc-editor.org/info/rfc4737 (accessed on 11 June 2023). [CrossRef]
93. Piratla, N.M.; Jayasumana, A.P.; Banka, T. On reorder density and its application to characterization of packet reordering. In Proceedings of the IEEE Conference on Local Computer Networks 30th Anniversary (LCN'05)l, Sydney, NSW, Australia, 17 November 2005; IEEE: Piscataway, NJ, USA, 2005.
94. Piratla, N.M. *A Theoretical Foundation, Metrics and Modeling of Packet Reordering and Methodology of Delay Modeling Using Inter-Packet Gaps*; Colorado State University: Fort Collins, CO, USA, 2005.
95. Wu, W.; Demar, P.; Crawford, M. Sorting Reordered Packets with Interrupt Coalescing. *Comput. Netw.* **2009**, *53*, 2646–2662. [CrossRef]
96. Evensen, K.; Kaspar, D.; Engelstad, P.; Hansen, A.F.; Griwodz, C.; Halvorsen, P. A network-layer proxy for bandwidth aggregation and reduction of IP packet reordering. In Proceedings of the 34th Annual IEEE Conference on Local Computer Networks, LCN 2009, Zurich, Switzerland, 20–23 October 2009; pp. 585–592.
97. Kaspar, D. Multipath aggregation of heterogeneous access networks. *ACM* **2012**, *4*, 27–28. [CrossRef]

98. Ramaboli, A.L.; Falowo, O.E.; Chan, A.H. Bandwidth aggregation in heterogeneous wireless networks: A survey of current approaches and issues. *J. Netw. Comput. Appl.* **2012**, *35*, 1674–1690. [CrossRef]
99. Narasiodeyar, R.M.; Jayasumana, A.P. Improvement in packet-reordering with limited re-sequencing buffers: An analysis. In 38th Annual IEEE Conference on Local Computer Networks, Sydney, Australia, 21–24 October 2013; pp. 416–424.
100. Latif, S.A.; Masud, M.H.; Anwar, F.; Alam, M.K. An investigation of scheduling and packet reordering algorithms for bandwidth aggregation in heterogeneous wireless networks. *Middle East J. Sci. Res.* **2013**, *16*, 1613–1623.
101. El-Atawy, A.; Duan, Q.; Al-Shaer, E. A novel class of robust covert channels using out-of-order packets. *IEEE Trans. Dependable Secur. Comput.* **2015**, *14*, 116–129. [CrossRef]
102. Anwar, F.; Masud, M.H.; Khan, B.U.I.; Olanrewaju, R.F.; Latif, S.A. Analysis of packet reordering delay for bandwidth aggregation in heterogeneous wireless networks. *IPASJ Int. J. Inf. Technol.* **2018**, *6*.
103. Piratla, N.M.; Jayasumana, A.P.; Bare, A.A. Reorder density (RD): A formal, comprehensive metric for packet reordering. In Proceedings of the *NETWORKING 2005. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems, 4th International IFIP-TC6 Networking Conference, Waterloo, Canada, 2–6 May 2005*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 78–89.
104. Jin, K.; Dong, D.; Li, C.; Huang, L.; Ma, S.; Fu, B. DancerFly: An Order-Aware Network-on-Chip Router On-the-Fly Mitigating Multi-path Packet Reordering. *Int. J. Parallel Program.* **2020**, *48*, 730–749. [CrossRef]