electronics

MDPI

Article

Joint Congestion Control Evaluation for MPTCP and MPQUIC over Multi-Link Backhauls with eMBB and mMTC-like Traffic

Roberto Picchi [1] **and Daniele Tarchi** [2,*]

[1] CNIT—Consorzio Nazionale Interuniversitario per le Telecomunicazioni, University of Florence Research Unit, Via Santa Marta 3, 50139 Florence, Italy; roberto.picchi@cnit.it
[2] Department of Information Engineering, University of Florence, Via Santa Marta 3, 50139 Florence, Italy
* Correspondence: daniele.tarchi@unifi.it

Abstract

Multi-link terrestrial backhauls create a shared transport environment in which heterogeneous multipath protocols compete for the same forwarding resources while reacting to congestion with different control logics. In this paper, we investigate this problem in a 5G Integrated Access and Backhaul (IAB) scenario where an IAB node aggregates traffic from multiple User Equipments (UEs) and forwards it toward the core network over two terrestrial backhaul paths. We focus on the coexistence of Multipath TCP (MPTCP) and Multipath QUIC (MPQUIC), evaluating how cross-protocol Congestion Control (CC) pairings affect performance. Specifically, all feasible BBR, CUBIC, and Reno cross-pairings are assessed under symmetric and asymmetric dual-backhaul conditions, considering Enhanced Mobile Broadband (eMBB) and dense low-rate traffic regimes representative of mMTC-like operation. The analysis considers throughput, Jain's fairness index, jitter, and packet loss to identify the trade-offs of each CC pairing. Results show that CC selection is a first-order design factor in MPTCP/MPQUIC coexistence over shared backhauls. No single pairing is uniformly optimal across all metrics: some configurations provide more balanced throughput sharing, others improve fairness, while the most favorable solutions for jitter do not necessarily maximize transport efficiency. These findings identify CC pairing as a tuning dimension for multi-link backhaul systems based on heterogeneous multipath transports.

Keywords: congestion control; integrated access and backhaul; multi-connectivity; multipath QUIC; multipath TCP; terrestrial backhaul

1. Introduction

Multi-path transport has emerged as a relevant enabler for exploiting path diversity in heterogeneous networks, where multiple interfaces and alternative routes can be used concurrently to improve throughput, resilience, and resource utilization. Among the available solutions, Multipath TCP (MPTCP) extends conventional TCP by allowing data to be delivered simultaneously over multiple network interfaces, so that a connection is no longer constrained to a single route but can instead aggregate bandwidth across all available links [1]. By coupling multiple parallel subflows into one logical transport connection, MPTCP can significantly improve throughput while making more effective use of heterogeneous resources. At the same time, it preserves backward compatibility with legacy TCP behavior, thus remaining interoperable with existing network infrastructures and endpoints [2].

Received: 23 March 2026
Revised: 15 April 2026
Accepted: 21 April 2026
Published: 23 April 2026

These properties are particularly attractive in environments where multiple paths are naturally available. In wireless systems, for example, devices commonly expose multiple access technologies, such as Wi-Fi and cellular, and can therefore benefit from traffic distribution across heterogeneous interfaces. This improves achievable data rates and increases resilience to link disruptions, thereby sustaining End-to-End (E2E) connectivity under mobility and fluctuating coverage conditions [3]. More generally, multipath operation also enhances robustness, since traffic can be shifted away from degraded or failed links as network conditions evolve [4]. An illustrative example of an MPTCP session using two parallel TCP subflows mapped onto distinct E2E network paths is shown in Figure 1.

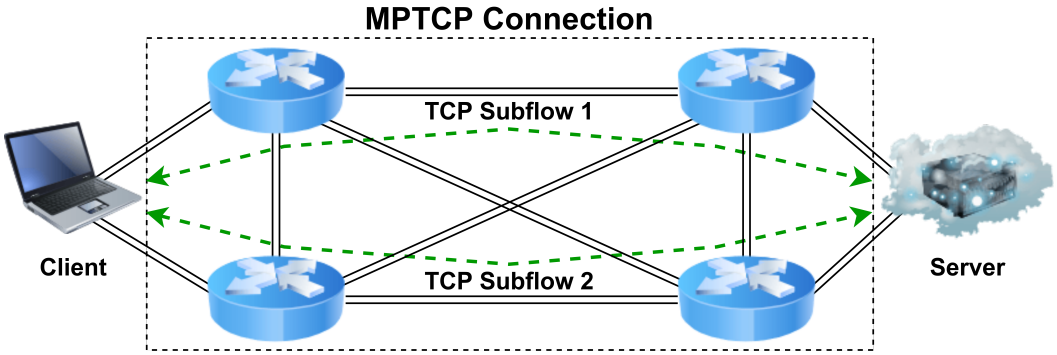

Figure 1. Example of an MPTCP session using two parallel TCP subflows mapped onto distinct E2E network paths.

In parallel with TCP-based multipath transport, the emergence of QUIC as a user-space transport protocol has introduced additional flexibility, including reduced connection–establishment latency, built-in security, and improved tolerance to network changes. Building on these features, Multipath QUIC (MPQUIC) augments QUIC [5] with native multipath capabilities, thus enabling concurrent path usage while avoiding kernel coupling and reducing exposure to TCP-specific middlebox interference. Similar to MPTCP, MPQUIC can distribute traffic across heterogeneous interfaces; moreover, it inherits QUIC-native mechanisms such as connection migration and encryption by default, which are particularly beneficial in mobile and wireless environments. For these reasons, MPQUIC is increasingly regarded as a promising complementary substrate for multi-connectivity, especially in scenarios characterized by frequent path variation, pronounced delay asymmetry, and intermittent connectivity [6,7].

In the multi-link terrestrial backhaul architecture considered in this work, these multipath transport capabilities are exploited at the backhaul level as well. More specifically, multiple User Equipments (UEs) connect to a 5G Integrated Access and Backhaul (IAB) node. The IAB node aggregates the access traffic and forwards it toward the core network over two concurrent terrestrial backhaul paths. The operating conditions of these paths may vary over time because of load fluctuations and transient queueing effects [8–10]. Such a dual-backhaul configuration is attractive because it can improve utilization and service continuity, but it also exposes the transport layer to shared-resource contention, queue buildup, and path-dependent variability. As a consequence, E2E performance depends not only on the availability of multiple paths, but also on the interaction among Congestion Control (CC), queueing dynamics, and protocol pacing across the two backhaul links.

This problem is aligned with the evolution of 5G architectures. Specifically, these architectures rely on multi-link terrestrial backhauls to improve capacity, robustness, and service continuity under variable load and channel conditions [11–13]. In these systems, UEs may connect directly to terrestrial base stations or indirectly through IAB nodes that aggregate access traffic and forward it over heterogeneous terrestrial backhaul options [12].

Although this multi-link structure is instrumental in improving resilience and resource utilization, it also requires effective coordination mechanisms to preserve Quality of Service (QoS) under diverse operational conditions [14,15]. In this context, multi-connectivity becomes a key architectural primitive and is explicitly supported by the 3GPP Access Traffic Steering, Switching, and Splitting (ATSSS) framework [16]. ATSSS enables multiple links to be used simultaneously or selectively to improve E2E throughput, reliability, and continuity by steering traffic to the most suitable path, splitting it across parallel accesses, or switching to a backup when link conditions require it [17]. At the same time, multi-link terrestrial backhauls may exhibit time-varying delay, throughput, and jitter due to shared-resource contention and transient buffering effects, which can trigger packet reordering and throughput degradation if not properly handled [18,19]. Coping with these effects requires scheduling and transport-layer mechanisms that can adapt to backhaul dynamics and coordinate transmissions over multiple paths in a timely and stable way [20]. In this respect, MPTCP and MPQUIC, by controlling multiple subflows within a single connection, represent promising transport substrates for reliable and high-performance multi-connectivity over multi-link terrestrial backhauls.

Beyond broadband-oriented traffic, the considered IAB architecture is also relevant under dense Massive Machine Type Communications (mMTC) operating conditions, where a large number of devices may generate short and sporadic transmissions that collectively stress both access and backhaul resources [21,22]. In 5G systems, mMTC is recognized as one of the three main service categories and is associated with very high device density, low-complexity terminals, and traffic patterns that can intensify signaling overhead, random-access contention, and resource-allocation challenges [23]. The experimental settings considered in this paper are not intended to reproduce a complete 3GPP mMTC deployment. However, the dense-user and mixed-traffic cases can be interpreted as mMTC-inspired conditions, which are useful for assessing how transport-layer CC affects throughput sharing, fairness, jitter, and packet loss when many low-rate sources concurrently feed a dual-link terrestrial backhaul.

From the Radio Access Network (RAN) viewpoint, IAB tightly couples access scheduling and backhaul forwarding, so that coordinating resources across these segments becomes a first-order design issue rather than a secondary optimization. Accordingly, a broad literature has investigated joint access/backhaul management under IAB constraints. For millimeter-wave (mmWave) systems, analytical models have been used to characterize access–backhaul partitioning and its impact on downlink performance [24], while New Radio (NR)-oriented solutions have explored semi-centralized control to coordinate topology-aware backhaul scheduling with access demand under practical constraints [25]. Building on this foundation, distributed local strategies have been proposed for per-slot scheduling in multi-hop mmWave IAB networks and have been further extended with flow-control-aware resource allocation to better track traffic dynamics [26]. In the same direction, learning-based forwarding and traffic-engineering approaches have recently been explored to improve adaptivity also in sub-6 IAB scenarios [27,28].

What remains comparatively less explored is how heterogeneous multipath transport protocols behave over dual terrestrial backhaul links when protocol coexistence directly affects congestion and queueing dynamics. This is the perspective adopted in the present paper, which focuses on MPTCP/MPQUIC operation in an IAB-based multi-link terrestrial scenario, where multiple UEs are served by a 5G IAB node that aggregates access traffic and forwards it toward a core-network server over two concurrent terrestrial backhaul links. Within this framework, we conduct an in-depth comparative study of CC behavior by evaluating cross-combinations of Bottleneck Bandwidth and Round-trip propagation time (BBR), CUBIC, and Reno across the MPTCP and MPQUIC stacks, and by quantifying

the resulting E2E throughput, fairness, and jitter under shared backhaul resources and time-varying terrestrial conditions.

More specifically, the main contributions of this work can be summarized as follows:

- A systematic experimental assessment of CC cross-combinations between MPTCP and MPQUIC over symmetric and asymmetric dual-link terrestrial backhaul configurations, providing a unified comparison under a common shared-resource setting.
- A comparative characterization of throughput, fairness, jitter, and packet loss under Enhanced Mobile Broadband (eMBB) and mMTC-inspired traffic conditions.
- The identification of the CC pairings that provide the most favorable trade-offs depending on whether the target objective is throughput efficiency, fairness, delay stability, or packet-loss behavior.

The remainder of the paper is organized as follows. Section 2 reviews the related work. Section 3 describes the system model and experimental methodology. Section 4 presents the obtained results. Finally, Section 5 concludes the paper and outlines future research directions.

2. Related Work

The literature related to multipath transport in integrated and heterogeneous networking environments can be grouped along multiple directions. A first set of works investigates the behavior of MPTCP and MPQUIC under realistic wireless and 5G conditions. A second line of research focuses on Software-Defined Networking (SDN)-driven architectures, routing strategies, and network-side control solutions to support multipath operation in dynamic topologies. A third body of work addresses IAB resource coordination, delay-aware scheduling, and congestion phenomena in access/backhaul-integrated systems. Finally, several transport-oriented studies examine the interaction between congestion control, scheduling, and heterogeneous links, including hybrid terrestrial and optical environments.

Experimental studies on practical 5G and mobile environments have clarified how multipath transport behaves under realistic and time-varying network conditions. An early empirical study on commercial 5G networks shows that MPTCP performance is strongly influenced by RAN variability and radio-level policies, thus providing useful guidance for evaluating TCP/MPTCP in real 5G settings [29]. More generally, recent works have further clarified the impact of heterogeneous wireless conditions on multipath protocols. In particular, ref. [30] discusses the practical challenges of deploying MPTCP over mmWave 5G and beyond, identifying path asymmetry, blockage sensitivity, and scheduler interactions among the main implementation issues. In addition, Ref. [31] proposes a packet scheduler for MPTCP in mobile networks, showing that packet-distribution decisions become a first-order factor when paths are unstable and heterogeneous. In the case of MPQUIC, Ref. [32] develops a Mobility-Aware Multipath Scheduler (MAMS), which explicitly exploits mobility awareness to improve path utilization and packet assignment in dynamic wireless scenarios. Moreover, Ref. [33] shows that learning-based scheduling combined with adaptive congestion control can improve throughput and delay in different multipath settings. Beyond scheduler-level adaptation, recent work has also explored AI-native protocol design. In particular, Ref. [34] shows that abstraction-based multi-agent reinforcement learning can enable the emergence of generalized wireless MAC protocols with adaptive and cooperative behavior.

SDN-based and controller-driven approaches have also been explored to support multipath operation in dynamic topologies. In [35], the authors introduce a Software-Defined Satellite Network (SDSN) framework building on the broader context of software-defined and integrated satellite networking solutions [15,36,37], in which a Ryu controller

installs OpenFlow routes for MPTCP subflows. Three path-selection strategies are implemented and compared, namely modified Dijkstra, Bandwidth-constrained least-Delay least-Cost Routing Algorithm (BDCRA), and Ant Colony Optimization (ACO). Experiments on Mininet indicate that ACO and BDCRA outperform Dijkstra, with ACO providing the highest average throughput at the price of increased computational complexity.

From the IAB perspective, several works have highlighted the importance of coordinated access/backhaul management and delay-aware resource allocation. In particular, Ref. [38] analyzes CC mechanisms in 3GPP IAB systems and shows that congestion at intermediate IAB nodes may arise when ingress traffic persistently exceeds egress backhaul capability, leading to prolonged queue buildup, increased waiting delay, and possible service degradation. This study emphasizes that such conditions cannot always be handled effectively by conventional short-term congestion reactions alone and therefore motivate longer-term control actions based on traffic splitting and topology-aware adaptation. Furthermore, Ref. [39] investigates delay-sensitive link scheduling in IAB networks with dynamic user demands by formulating the problem as a Markov Decision Process (MDP) for a general single-donor topology. Their results show that jointly accounting for routing, access/backhaul multiplexing, heterogeneous delay requirements, and traffic dynamics is essential to reduce packet delay while maintaining service satisfaction. Similarly, Ref. [40] provides a recent survey of delay-oriented scheduling policies for 5G/6G IAB systems, highlighting that delay becomes a central design metric in multi-hop IAB because of spectrum sharing, half-duplex constraints, and the fact that only a subset of access and backhaul links can be scheduled simultaneously.

The interaction between congestion control, scheduling, and heterogeneous paths has also been studied in transport-oriented works. In [41], BBR is analyzed together with MPTCP through both a physical testbed and Mininet emulation, varying the RTT asymmetry between two subflows and considering multiple schedulers. The results indicate that BBR outperforms CUBIC and Binary Increase Congestion control (BIC) when latency asymmetry is severe and that `minRTT` is generally the most robust scheduler. At the same time, overly large receiver buffers can significantly degrade goodput beyond roughly 20 ms of RTT difference, highlighting head-of-line blocking and non-trivial interactions between congestion control and scheduling. A recent experimental study on practical 5G eMBB networks compares the behavior of CUBIC TCP and TCP BBR using real 5G devices, showing that their throughput can exhibit minute-scale fluctuations and that coexistence may lead to severe unfairness, with large performance gaps that persist for non-negligible intervals [42]. The authors relate these long-term phases to deep buffering effects in 5G elements and to the resulting dynamics of BBR's estimation and draining mechanisms. Along the same line, Ref. [43] investigates why empirical studies report conflicting results for BBR when it competes with other loss-based CC mechanisms, showing that measured fairness can be strongly affected by convergence time in deep-buffered paths. Using controlled testbed experiments across different RTTs, link rates, queue sizes, and numbers of competing flows, that study validates that the proposed model predicts BBR's throughput share with low error, thus providing a principled explanation for the observed unfairness regimes and their dependence on buffer size and probing dynamics.

Additional studies have considered more specialized heterogeneous links. In [44], a cross-layer analytical framework evaluates multiple TCP variants over hybrid FSO/RF links affected by turbulence, pointing errors, and orbital geometry. The study shows that throughput is strongly influenced by parameters such as segment size and zenith angle; specifically, CUBIC is generally the best performer in many conditions, while Hybla is preferable when RTT and packet loss are high. In a complementary manner, Ref. [45] quantifies the benefits of link-layer Incremental Redundancy Hybrid Automatic Repeat

reQuest (IR-HARQ), showing that incremental redundancy can substantially improve TCP throughput over turbulent FSO channels. The analysis compares CUBIC and Hybla, arguing that the preferable CC choice depends on the dominant error regime and on the level of Internet congestion, while also discussing the combined impact of turbulence and pointing-jitter-induced misalignment.

Overall, these contributions outline the current state of the art on multipath transport in heterogeneous environments. They cover real-world 5G measurements, SDN-driven and controller-based support for multipath management, IAB-oriented scheduling and congestion analysis, transport-level studies on MPTCP with BBR and other CC algorithms, and cross-layer analyses of FSO and hybrid FSO/RF systems. However, a unified view of how MPTCP and MPQUIC interact under different CC choices over dual terrestrial backhauls is still missing, especially when heterogeneous multipath transports concurrently share the same backhaul resources and compete for the same bottleneck capacity. To address this gap, we present a comparative evaluation of MPTCP and MPQUIC under all feasible cross-combinations of BBR, CUBIC, and Reno, quantifying the resulting throughput, fairness, and jitter dynamics over a dual-link terrestrial backhaul scenario under both eMBB and mMTC-inspired traffic conditions.

3. System Model and Experimental Methodology

3.1. Scenario Definition

The system model is designed to isolate the impact of CC interactions between MPTCP and MPQUIC in a dual-link terrestrial backhaul environment. Accordingly, the proposed model intentionally abstracts away several aspects of a full 5G system, including detailed RAN procedures, radio scheduling, HARQ, and access-side signaling dynamics. Even though these elements may affect deployment realism, they are not the focus of the present study. The objective here is instead to isolate the backhaul-side transport effects induced by shared-resource contention and by the coexistence of heterogeneous multipath protocols under different CC pairings. In this sense, the adopted abstraction should be interpreted as a controlled transport-level reference scenario, rather than as a complete E2E emulation of all 5G mechanisms.

To this end, we consider a 5G IAB-based scenario in which multiple UEs generate traffic toward a remote server, while an intermediate IAB node aggregates the access traffic and forwards it through two concurrent terrestrial backhaul links. The set of UEs is evenly divided into MPTCP-enabled and MPQUIC-enabled devices, so that the coexistence between the two multipath transport families can be evaluated under shared backhaul operation. The considered reference scenario is illustrated in Figure 2.

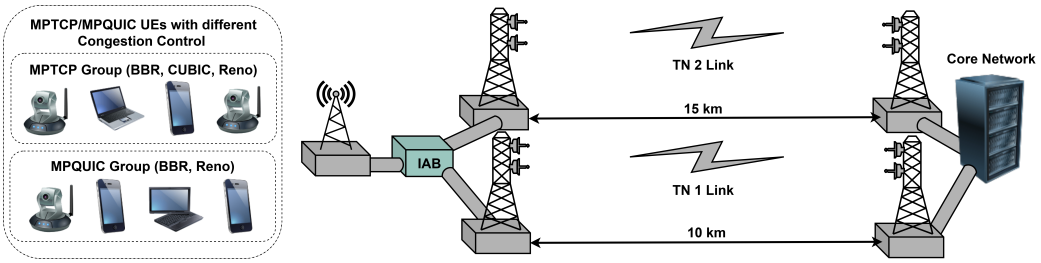

Figure 2. Multi-link terrestrial backhaul scenario with MPTCP-enabled and MPQUIC-enabled UEs and an IAB node aggregating traffic over two terrestrial backhaul links toward the core network.

The adopted abstraction focuses on the backhaul-side transport behavior. More specifically, the model is not intended to reproduce all radio-access procedures in detail; instead, it captures the effects of shared-resource contention, transient queue buildup, and path-dependent variability on the E2E performance of MPTCP and MPQUIC flows.

This level of abstraction is suitable for comparing CC pairings, since it allows the impact of CC selection to be assessed without introducing unrelated sources of variability.

UE traffic is conveyed through the IAB node, which aggregates the access-side flows and relays them toward the remote server over two terrestrial backhaul links. This dual-backhaul structure emulates a shared forwarding environment in which both paths may jointly contribute to transport delivery or expose different levels of utility because of latency, throughput, or availability differences. In this way, the topology captures the shared backhaul conditions under which the coexistence between MPTCP and MPQUIC is evaluated.

3.2. Emulation Framework

The considered topology is instantiated in Python 3.8.10 through the Mininet network emulator, which provides a controlled and reproducible environment for transport-layer experimentation. By relying on Linux network namespaces and virtual links, Mininet allows each emulated node to run the unmodified Linux network stack together with standard user-space applications [46]. This makes it well suited to the present study, where the objective is to compare MPTCP/MPQUIC CC interactions under repeatable operating conditions while preserving realistic protocol behavior. In the adopted setup, the emulated hosts are based on Ubuntu and include the protocol support required to execute the considered MPTCP and MPQUIC sessions.

3.3. Backhaul and Traffic Configurations

To evaluate the impact of backhaul heterogeneity on transport-layer coexistence, the experimental campaign considers two dual-backhaul configurations. The first corresponds to a symmetric setting, in which both terrestrial backhaul links have the same length, equal to 10 km. The second corresponds to an asymmetric setting, where the two links have different lengths, namely 10 km and 15 km. This comparison allows the behavior of the considered transport solutions to be assessed under both balanced and heterogeneous backhaul conditions, while leaving the overall network architecture and traffic-generation framework unchanged.

Furthermore, in addition to the eMBB experiments, the campaign also includes a denser access regime that is described in the following as an mMTC-inspired scenario. In this context, the term mMTC-inspired is used to denote a high-density set of low-rate devices simultaneously sharing the same dual terrestrial backhaul, so as to stress contention, aggregation, and transport coexistence.

Within the considered dual-backhaul framework, the two terrestrial paths may be exploited concurrently or with different relative weights, depending on the traffic conditions and on the characteristics of the available links. When the two backhaul paths expose comparable delay and throughput conditions, both can contribute jointly to transport delivery so as to improve aggregate capacity. Conversely, when one path becomes less favorable, for example because of higher delay variability or transient queue buildup, traffic usage may become less balanced, with the more stable link carrying a larger fraction of the load while the other still contributes to robustness or additional capacity. In this work, this behavior is not studied from the perspective of scheduler design; rather, it provides the operating context in which the coexistence between MPTCP and MPQUIC and the effect of different CC pairings are assessed.

3.4. Transport Configuration and CC Settings

On the transport side, MPTCP-capable hosts establish an initial TCP subflow and then attempt to create additional subflows between the same endpoints when multiple interfaces are available. MPTCP operation requires that the remote peer correctly recog-

nizes and acknowledges the extended TCP option fields used for subflow negotiation. If such a negotiation fails, the connection transparently falls back to standard single-path TCP. In parallel, MPQUIC-enabled hosts exploit QUIC-based multipath at user space, enabling concurrent path usage without requiring kernel-level QUIC support. Within this setup, packet transmission over the available backhaul subflows follows the transport-level multipath operation supported by the considered protocol stack.

The MPTCP implementation relies on the default in-kernel scheduler, implemented through eBPF (extended Berkeley Packet Filter) `struct_ops` [47,48]. This scheduler follows a latency-oriented policy commonly referred to as `MinRTT`, whereby the subflow with the lowest estimated `RTT` is generally preferred whenever it has immediate sending opportunity, i.e., when its transmission is not temporarily limited by CC dynamics or other local constraints. When the preferred subflow cannot sustain further transmissions, traffic is shifted to the remaining available subflows. As a result, the scheduler tends to fill the sending opportunity of the minimum-`RTT` path first, while still allowing other paths to contribute when needed. In the present work, this mechanism is not itself the object of analysis; rather, it provides the baseline path-selection behavior under which the impact of different CC pairings is assessed.

TCP congestion control regulates the sender's flight size and sending rate to match the available path capacity while preventing persistent queue growth. In classic loss-based TCP, congestion is inferred primarily from loss, via duplicate ACKs or retransmission timeouts, with the congestion window acting as the main actuator that limits in-flight data. The available tools used to model traffic, namely `Iperf` [49] and `Picoquic` [50], make it possible to model the following CC strategies:

- **Reno:** TCP Reno is the canonical Additive-Increase/Multiplicative-Decrease (AIMD) controller. It probes for additional capacity by increasing *cwnd* during *slow start* with an approximately exponential growth and then in *congestion avoidance* with approximately linear growth, roughly one MSS per RTT when ACK-clocked. When congestion is detected through loss, Reno reduces the sending pressure via multiplicative decrease, and uses *fast retransmit/fast recovery* to recover promptly from single losses without re-entering slow start. In practice, timer-driven loss detection Retransmission Timeout (RTO) and its estimation strongly affect Reno's behavior under high delay variation or intermittent loss, since an RTO triggers a more conservative restart than duplicate-ACK-based recovery [51].
- **CUBIC:** CUBIC preserves the Reno-style loss signal and multiplicative decrease, but replaces Reno's linear window growth with a cubic growth law in time since the last congestion event. Intuitively, CUBIC grows slowly near the previous operating point, improving stability, then accelerates to efficiently utilize large Bandwidth-Delay Products (BDP), and includes a Reno-friendly region to remain reasonably fair when competing with classic Reno flows. Modern specifications also incorporate deployment-driven refinements, e.g., improved convergence behavior and startup guidance, reflecting its widespread adoption as the default TCP congestion controller in major stacks [52].
- **BBR:** BBR departs from loss-as-congestion by explicitly modeling the path using online estimates of bottleneck bandwidth (delivery rate) and minimum RTT propagation component. It then controls both the pacing rate (send rate) and the allowed in-flight volume, a BDP-scaled *cwnd* target, to operate near the optimum, i.e., high throughput with bounded queues. Rather than waiting for loss to signal congestion, BBR periodically probes for available bandwidth and for RTT inflation (queue growth) through cyclic gain patterns, e.g., *ProbeBW* and *ProbeRTT*. Recent specifications, such as BBRv2, also integrate more explicit reactions to loss/ECN and fairness considerations,

aiming to improve coexistence with loss-based flows and reduce persistent queue pressure across a broader set of bottleneck conditions [53].

3.5. Evaluation Methodology and Reported Metrics

The simulation campaign was executed on a machine equipped with an Intel Core i7-10750 CPU at 2.6 GHz, 16384 MB of RAM, and Ubuntu 20.04.6 LTS. The considered scenarios were instantiated in the Mininet network emulator [46], with the topology and experiments defined through Python scripts. Traffic generation was performed using protocol-specific tools: `Iperf` in multipath mode [49] for MPTCP, relying on the native Linux-kernel MPTCP stack [54], and `Picoquic` [50] in multipath mode for MPQUIC.

Within this setup, all feasible cross-combinations of the selected CC algorithms across the MPTCP and MPQUIC stacks were evaluated. For each pairing, throughput, fairness, jitter, and packet-loss statistics were collected over the dual terrestrial backhaul. For clarity, the reported results include: (i) per-link throughput distributions for MPTCP and MPQUIC on each backhaul link; (ii) jitter distributions comparing the two backhaul links; and (iii) protocol-specific jitter distributions on each backhaul link separately. In addition, fairness is summarized through Jain's index, while throughput and packet-loss statistics are reported in dedicated summary tables for each scenario. The throughput analysis highlights how each CC pairing affects not only aggregate utilization but also the sharing of traffic between MPTCP and MPQUIC over the two terrestrial backhaul links, whose configured capacities are reported in Table 1. Furthermore, in the scenarios considered, we assumed a packet size of 1500 bytes for eMBB traffic and 100 bytes for mMTC-inspired traffic [55], while the 8 ms and 10 ms [56] values represented in Table 1 are the emulated per-path latency settings in the controlled backhaul abstraction. The latency values of TN1 and TN2 represent baseline emulated path delays. However, the effective packet latency may vary with packet size and traffic load because of serialization and queueing effects.

Table 1. Configured backhaul capacities for the considered scenario.

Backhaul Links	Capacities [Mbit/s]	Latencies [ms]
TN 1	501.74	8
TN 2	436.43	10

In all the scenarios considered below, the traffic generated is assumed to be uplink traffic, i.e., from the UEs to the core network. In the mMTC-inspired scenario, throughput is not the only relevant metric and may become secondary to delay stability, since a large number of devices generate sporadic low-rate traffic whose service quality depends more on predictable delivery than on high per-device throughput [57]. For this reason, in the corresponding analysis, particular attention is paid to the CC combinations that provide lower jitter in addition to acceptable throughput behavior.

4. Results

This section presents the results obtained under the experimental settings described in Section 3. For clarity, the discussion is organized according to the considered backhaul and traffic conditions. We first analyze the symmetric dual-backhaul setting under eMBB and mMTC-inspired traffic regimes, and then move to the asymmetric case. Finally, a mixed scenario is discussed to provide an additional validation point for the observed trends.

4.1. Symmetric Dual-Backhaul: eMBB Scenario

We first consider the symmetric dual-backhaul setting, in which both terrestrial backhaul links have length equal to 10 km and therefore expose balanced path conditions.

In this scenario, 12 UEs generate eMBB traffic at 100 Mbps per device, with half of the users employing MPTCP and the other half MPQUIC. This configuration is used to assess how the considered CC pairings behave when path asymmetry is not the dominant factor and the coexistence between the two protocol families is mainly shaped by their transport dynamics over shared backhaul resources.

Across the considered CC pairings, the symmetric setting shows that the two terrestrial backhaul links are utilized in a largely balanced way, so that the main differences emerge from the interaction between the selected CC algorithms and the two transport stacks rather than from path asymmetry itself.

As shown in Figure 3a, under the BBR-BBR pairing MPTCP achieves consistently higher throughput than MPQUIC on both terrestrial backhaul links. In particular, MPTCP operates around $\mu = 289.89$ Mbps (TN1) and $\mu = 300.76$ Mbps (TN2), while MPQUIC achieves lower mean rates of $\mu = 206.83$ Mbps (TN1) and $\mu = 195.97$ Mbps (TN2), confirming a persistent throughput advantage for the TCP-based stack even when both protocols adopt BBR. Figure 3b reports the protocol independent jitter distributions for the two backhaul links, which largely overlap, suggesting comparable delay variability across the two terrestrial paths with only minor differences in mean and spread. Figure 3c,d isolate protocol-specific jitter on TN1 and TN2, respectively. In both cases, MPQUIC concentrates at lower jitter values (around $\mu \approx 2$ ms) but exhibits a heavier tail due to occasional high-jitter outliers, while MPTCP is shifted to higher jitter levels in the $\mu \approx 3.6$–3.7 ms interval, with a comparatively tighter distribution. Applying BBR on both stacks, the two terrestrial links show similar jitter characteristics, while protocol-specific implementation and pacing dynamics primarily affect the jitter operating point and tail behavior.

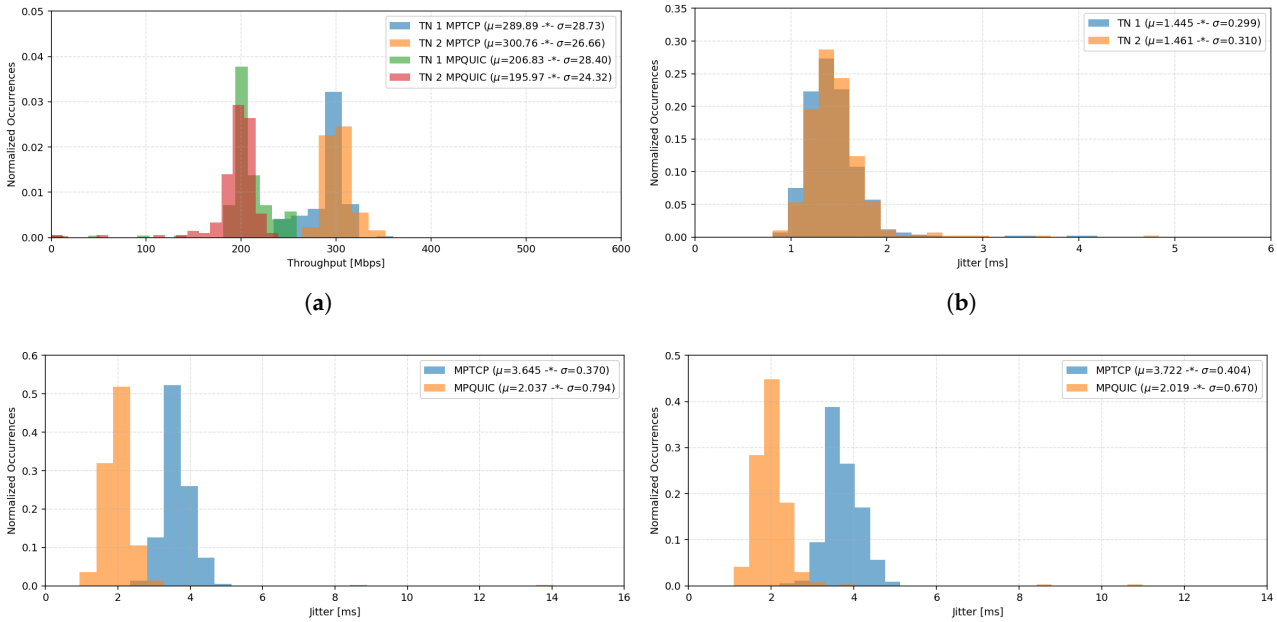

Figure 3. MPTCP and MPQUIC throughput and jitter distributions in symmetric eMBB scenario under the BBR-BBR CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

In Figure 4a, the throughput distributions for the BBR-CUBIC pairing show comparable operating regions on both terrestrial backhaul links, with largely overlapping distributions across TN1 and TN2. In general, MPQUIC achieves slightly higher mean rates than MPTCP on both links (TN1: $\mu = 276.36$ Mbps vs. $\mu = 220.69$ Mbps; TN2:

μ = 254.98 Mbps vs. μ = 241.67 Mbps), indicating a modest cross-protocol imbalance that remains consistent across the two backhaul paths. In Figure 4b, the protocol independent jitter distributions for TN1 and TN2 are almost indistinguishable (μ = 3.339 ms and μ = 3.332 ms, respectively), suggesting similar link-level delay variability. Figure 4c,d isolate protocol-specific jitter on TN1 and TN2: MPTCP remains tightly concentrated around low jitter (TN1: μ = 4.341 ms; TN2: μ = 4.330 ms), whereas MPQUIC exhibits a much broader, heavy-tailed distribution with markedly higher mean values (TN1: μ = 14.722 ms; TN2: μ = 15.303 ms) and outliers extending beyond 40 ms. This consistent pattern across both terrestrial links indicates that, under the BBR-CUBIC pairing, MPQUIC experiences substantially higher inter-arrival variability than MPTCP even when the two backhaul paths exhibit similar link-level jitter.

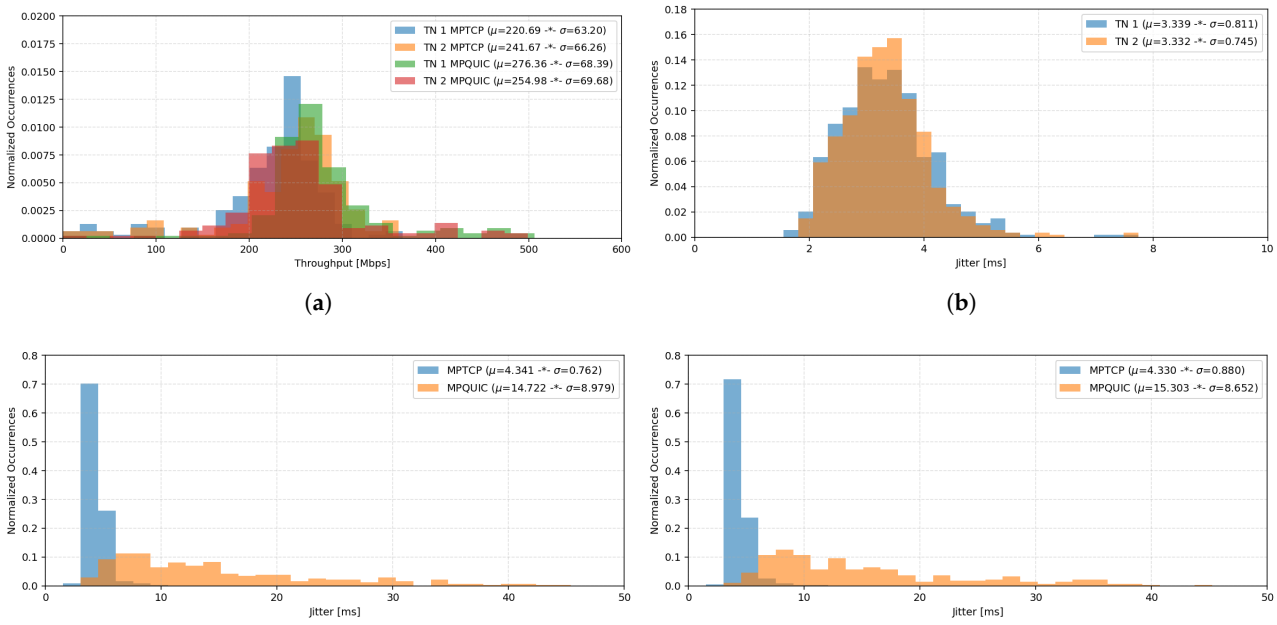

Figure 4. MPTCP and MPQUIC throughput and jitter distributions in symmetric eMBB scenario under the BBR-CUBIC CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

In Figure 5a, the throughput distributions for the CUBIC-BBR pairing show a consistent advantage for MPTCP over MPQUIC on both terrestrial backhaul links. In particular, MPTCP operates around μ = 272.29 Mbps on TN1 and μ = 269.71 Mbps on TN2, while MPQUIC is centered at lower rates with μ = 218.55 Mbps on TN1 and μ = 219.90 Mbps on TN2. The close alignment of each protocol across TN1 and TN2 indicates a largely symmetric utilization of the two terrestrial backhaul links, without a marked link preference. Figure 5b reports the protocol independent jitter comparison between TN1 and TN2. The two distributions largely overlap, with comparable averages (μ = 1.563 ms on TN1 and μ = 1.572 ms on TN2), while TN2 exhibits a slightly wider spread, suggesting marginally higher variability in inter-arrival dynamics on that link. Figure 5c,d isolate protocol-specific jitter on TN1 and TN2, respectively. On TN1 (Figure 5c), MPQUIC shows a lower average jitter (μ = 2.287 ms) than MPTCP (μ = 4.011 ms), but with a broader dispersion, indicating occasional higher-jitter events despite a lower typical operating region. A similar behavior is observed on TN2 (Figure 5d), where MPQUIC remains lower on average (μ = 2.288 ms) but more variable, whereas MPTCP is concentrated at higher jitter values (μ = 3.772 ms) with a tighter distribution. This pairing yields comparable link-level jitter across TN1/TN2,

while highlighting a trade-off where MPQUIC achieves lower typical jitter but with heavier tails than MPTCP on both terrestrial backhaul links.

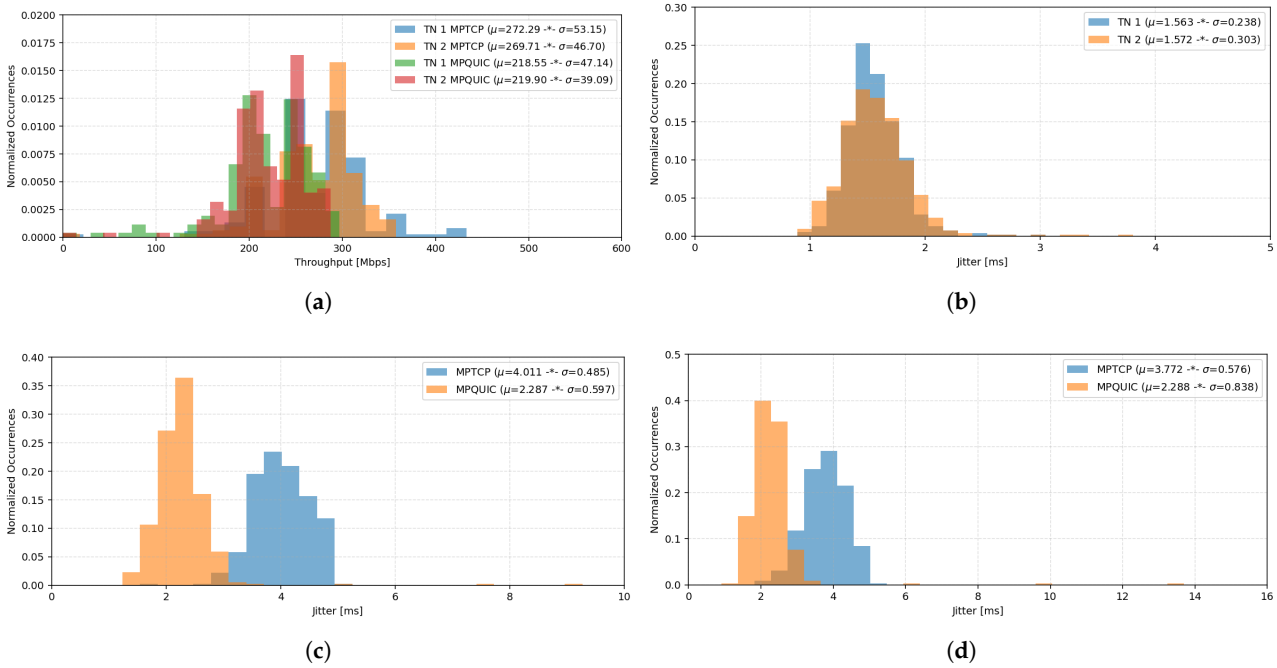

Figure 5. MPTCP and MPQUIC throughput and jitter distributions in symmetric eMBB scenario under the CUBIC-BBR CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

In Figure 6a, the CUBIC-CUBIC pairing yields a relatively stable sharing across the two terrestrial backhaul links. MPTCP operates at slightly higher throughput than MPQUIC on both links, with mean rates of $\mu = 266.00$ Mbps (TN1) and $\mu = 269.51$ Mbps (TN2) for MPTCP, versus $\mu = 225.27$ Mbps (TN1) and $\mu = 227.24$ Mbps (TN2) for MPQUIC, while both protocols remain well distributed over the two paths. Figure 6b reports the protocol independent jitter comparison between the two backhaul links, showing closely overlapping distributions with comparable averages (TN1: $\mu = 2.662$ ms, TN2: $\mu = 2.692$ ms), indicating that link-level delay variability is similar across the two terrestrial segments. Figure 6c,d isolate protocol-specific jitter on TN1 and TN2, respectively. On both links, MPTCP exhibits a compact distribution centered around ≈ 4 ms (TN1: $\mu = 4.225$ ms; TN2: $\mu = 4.005$ ms), whereas MPQUIC shows higher average jitter and markedly broader, heavy-tailed distributions (TN1: $\mu = 5.179$ ms; TN2: $\mu = 5.684$ ms), highlighting larger inter-arrival variability for MPQUIC despite comparable link-level conditions.

In Figure 7a, the throughput distributions for the Reno-BBR pairing show a consistent advantage of MPTCP over MPQUIC on both terrestrial backhaul links. In particular, MPTCP achieves mean rates of $\mu = 277.78$ Mbps on TN1 and $\mu = 247.92$ Mbps on TN2, while MPQUIC remains substantial with $\mu = 210.93$ Mbps on TN1 and $\mu = 225.43$ Mbps on TN2, indicating a moderate but stable cross-protocol imbalance rather than starvation. Figure 7b compares protocol independent jitter across TN1 and TN2 and shows largely overlapping distributions ($\mu = 1.552$ ms on TN1 versus $\mu = 1.569$ ms on TN2), with TN2 exhibiting a slightly broader spread. Figure 7c,d isolate protocol-specific jitter on TN1 and TN2, respectively. On both links, MPQUIC features lower average jitter ($\mu = 2.277$ ms on TN1 and $\mu = 2.180$ ms on TN2) but higher dispersion, whereas MPTCP is shifted to higher jitter values ($\mu = 3.775$ ms on TN1 and $\mu = 3.726$ ms on TN2) with a more concentrated core distribution. This suggests that, under Reno-BBR, MPQUIC attains smoother average

inter-arrival timing but is more exposed to sporadic variability, while MPTCP exhibits more stable jitter around a higher operating level.

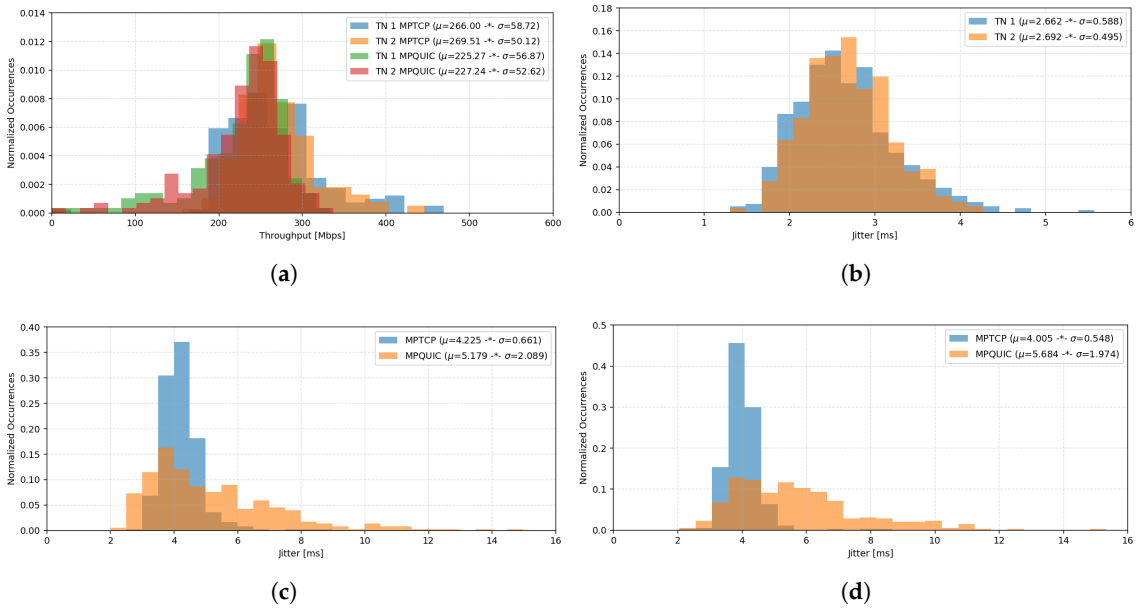

Figure 6. MPTCP and MPQUIC throughput and jitter distributions in symmetric eMBB scenario under the CUBIC-CUBIC CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

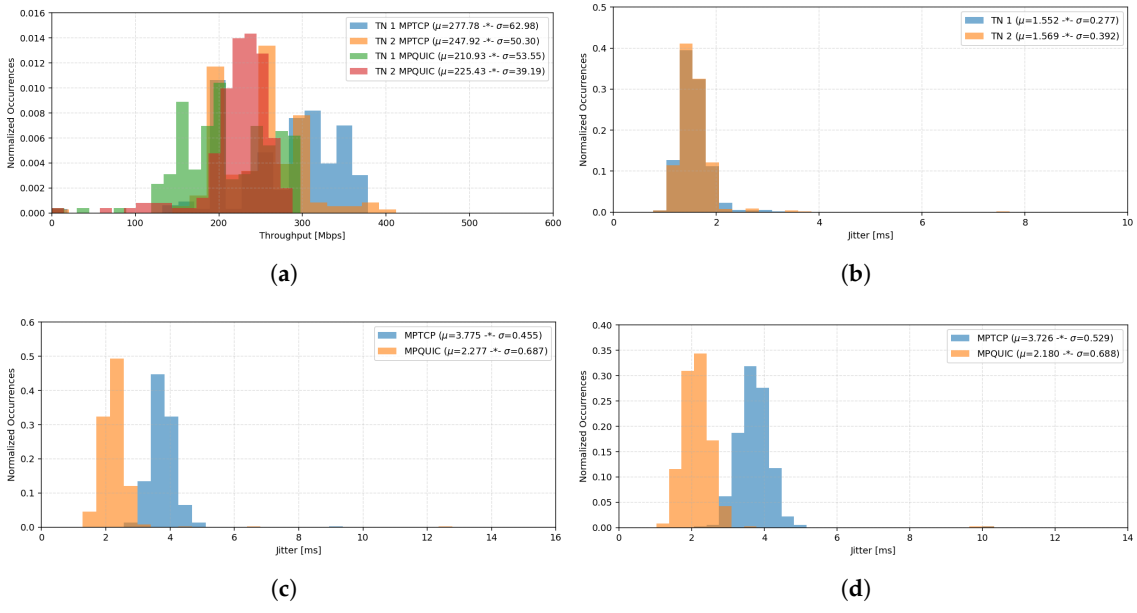

Figure 7. MPTCP and MPQUIC throughput and jitter distributions in symmetric eMBB scenario under the Reno-BBR CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

In Figure 8a, the throughput distributions for the Reno-CUBIC pairing show a moderate cross-protocol imbalance that depends on the backhaul link. On TN1, MPTCP and MPQUIC operate in partially overlapping regions, with MPTCP having slightly higher mean throughput ($\mu = 255.57$ Mbps) than MPQUIC ($\mu = 241.50$ Mbps). On TN2, the separation becomes clearer: MPTCP shifts to a higher-rate region ($\mu = 291.67$ Mbps), whereas

MPQUIC concentrates at lower rates ($\mu = 205.06$ Mbps), indicating reduced MPQUIC utilization of the second terrestrial backhaul link under this pairing. Figure 8b reports protocol independent jitter at the link level and shows largely overlapping distributions, with comparable mean jitter on TN1 ($\mu = 2.737$ ms) and TN2 ($\mu = 2.787$ ms), suggesting similar link-level delay variability when both protocols are considered jointly. Figure 8c,d separate protocol-specific jitter on TN1 and TN2, respectively. On TN1 (Figure 8c), MPQUIC exhibits higher and more dispersed jitter ($\mu = 5.732$ ms) than MPTCP ($\mu = 4.268$ ms), with a heavier right tail. The same trend is amplified on TN2 (Figure 8d), where MPQUIC shows a markedly broader, heavy-tailed distribution ($\mu = 6.787$ ms), while MPTCP remains tightly concentrated around lower jitter values ($\mu = 3.960$ ms). The Reno-CUBIC configuration yields reasonably comparable link-level jitter across TN1 and TN2. However, it induces substantially higher protocol-level jitter for MPQUIC, especially on TN2, together with a more pronounced throughput gap on that link.

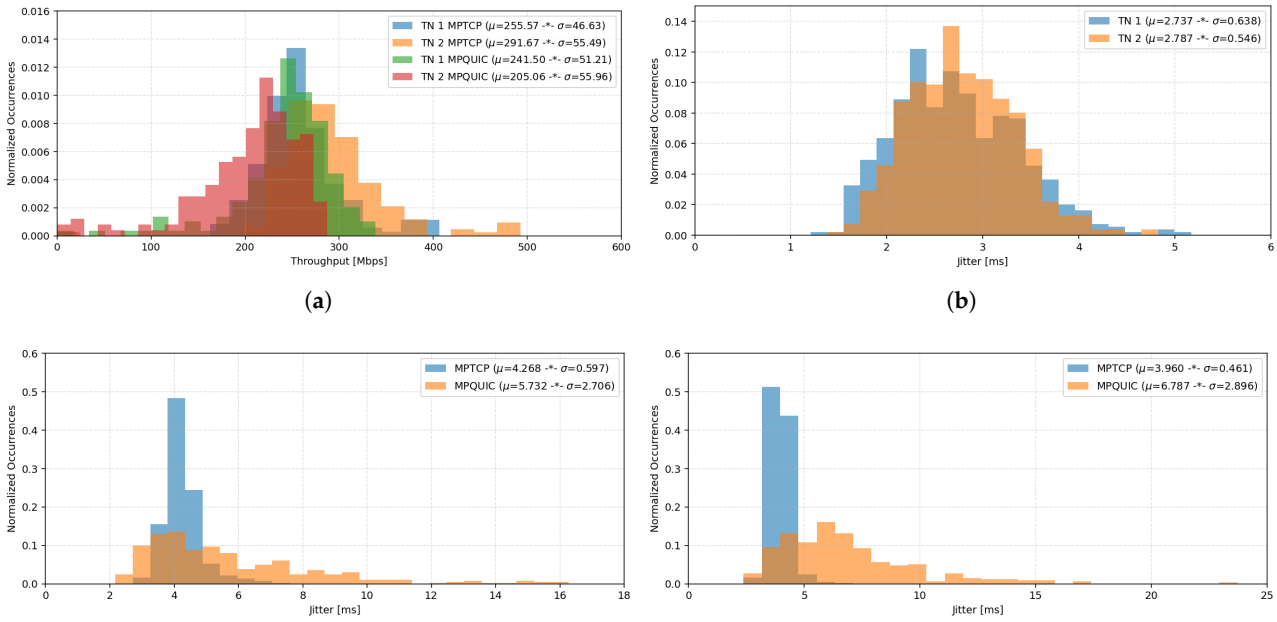

Figure 8. MPTCP and MPQUIC throughput and jitter distributions in symmetric eMBB scenario under the Reno-CUBIC CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

Beyond the per-pairing distributions, Tables 2 and 3 summarize the fairness and mean jitter trends observed across the considered CC combinations.

Table 2. Jain's fairness index between MPTCP and MPQUIC in Symmetric eMBB Scenario computed from the mean throughput values reported in panels (a) of the figures from Figures 3–8.

MPTCP/MPQUIC CC	TN1	TN2	Total
BBR-BBR	0.9728	0.9574	0.9655
BBR-CUBIC	0.9876	0.9993	**0.9952**
CUBIC-BBR	0.9882	0.9898	0.9890
CUBIC-CUBIC	**0.9932**	**0.9928**	0.9930
Reno-BBR	0.9816	0.9977	0.9915
Reno-CUBIC	0.9992	0.9705	0.9898

Table 3. Summary of the mean jitter values extracted from Figures 3–8 for the symmetric dual-link backhaul eMBB scenario. Boldface highlights the CC pairings providing the most favorable fairness performance.

MPTCP/MPQUIC CC	TN1	TN2	TN1 MPTCP	TN1 MPQUIC	TN2 MPTCP	TN2 MPQUIC
BBR-BBR	1.445	1.461	3.645	2.037	3.722	2.019
BBR-CUBIC	3.339	3.332	4.341	14.722	4.330	15.303
CUBIC-BBR	**1.563**	**1.572**	**4.011**	**2.287**	**3.772**	**2.288**
CUBIC-CUBIC	2.662	2.692	4.225	5.179	4.005	5.684
Reno-BBR	**1.552**	**1.569**	**3.775**	**2.277**	**3.726**	**2.180**
Reno-CUBIC	2.737	2.787	4.268	5.732	3.960	6.787

The fairness results in Table 2 show that two CC pairings emerge as the most favorable, depending on the adopted fairness objective. If fairness is evaluated on the aggregate throughput over the two terrestrial backhaul links, BBR-CUBIC provides the most balanced mean-throughput sharing between MPTCP and MPQUIC, achieving the highest overall Jain's index while also remaining close to one on each individual link. Conversely, if the goal is to maximize per-link fairness consistently on both backhaul links, CUBIC-CUBIC is the best option, yielding the highest indices on TN1 and TN2, with a slightly lower but still very high aggregate fairness.

From a jitter-oriented perspective, Table 3 indicates that the most favorable CC pairings in the symmetric dual-link backhaul scenario are Reno-BBR and CUBIC-BBR. In particular, Reno-BBR provides the lowest mean jitter across TN1 and TN2, both at link level and at protocol-specific level, while CUBIC-BBR exhibits very similar performance and confirms the same overall trend. By contrast, pairings such as BBR-CUBIC and Reno-CUBIC induce substantially larger jitter dispersion, especially for MPQUIC. For this reason, the following mMTC-inspired analysis focuses on Reno-BBR and CUBIC-BBR, as the two configurations that emerge as the most representative jitter-efficient solutions in the symmetric eMBB scenario.

These results highlight that throughput fairness and delay stability are not necessarily aligned. A pairing that equalizes the mean throughput across protocols may still exhibit larger jitter dispersion for one protocol or on one backhaul link, whereas other configurations may provide more stable inter-arrival behavior despite a slightly lower throughput balance.

However, Tables 4 and 5 confirm that the symmetric eMBB case does not align maximum throughput with fairness. While the highest overall throughput is reached only marginally by Reno-CUBIC, fairness points instead to BBR-CUBIC at aggregate level and to CUBIC-CUBIC when per-link balance is emphasized. Hence, in this scenario, the throughput-maximizing pairing is not the fairness-maximizing one.

Table 6 complements the throughput, fairness, and jitter analysis by showing that packet-loss behavior does not fully overlap with the previously identified throughput-oriented trends. In particular, the lowest packet-loss levels are observed for BBR-BBR, CUBIC-BBR, and, on the MPQUIC side, also Reno-BBR, whereas pairings such as BBR-CUBIC, CUBIC-CUBIC, and especially Reno-CUBIC exhibit clearly higher MPTCP loss levels. This means that the configurations previously highlighted for fairness or jitter are not necessarily those minimizing packet loss. In this respect, CUBIC-BBR and Reno-BBR remain particularly attractive, since they were already identified as jitter-efficient solutions and also maintain relatively contained loss values, thus representing a good overall compromise. Conversely, BBR-CUBIC, although the most favorable from the aggregate fairness standpoint, achieves this benefit at the price of a more pronounced packet-loss cost on the MPTCP side. Overall, Table 6 further confirms that no single CC

pairing is uniformly optimal across all metrics, and that the final preference depends on whether the design target prioritizes fairness, jitter stability, or loss containment.

Table 4. Summary of MPTCP throughput statistics for the symmetric eMBB scenario.

Path	CC	Min	Max	μ	σ
TN1	BBR-BBR	242.81	360.30	289.89	28.73
	BBR-CUBIC	5.79	364.72	220.69	63.20
	CUBIC-BBR	149.18	433.56	272.29	53.15
	CUBIC-CUBIC	178.41	469.01	266.00	58.72
	Reno-BBR	149.79	378.45	277.78	62.98
	Reno-CUBIC	169.01	406.45	255.57	46.63
TN2	BBR-BBR	264.21	351.95	300.76	26.66
	BBR-CUBIC	9.16	360.51	241.67	66.26
	CUBIC-BBR	130.37	357.63	269.71	46.70
	CUBIC-CUBIC	163.89	447.85	269.51	50.12
	Reno-BBR	150.18	412.13	247.92	50.30
	Reno-CUBIC	210.26	492.55	291.67	55.49

Table 5. Summary of MPQUIC throughput statistics for the symmetric eMBB scenario.

Path	CC	Min	Max	μ	σ
TN1	BBR-BBR	39.71	258.43	206.83	28.40
	BBR-CUBIC	44.76	506.19	276.36	68.39
	CUBIC-BBR	44.04	296.55	218.55	47.14
	CUBIC-CUBIC	31.66	332.84	225.27	56.87
	Reno-BBR	40.24	297.65	210.93	53.55
	Reno-CUBIC	43.47	338.72	241.50	51.21
TN2	BBR-BBR	54.37	239.07	195.97	24.32
	BBR-CUBIC	51.61	499.01	254.98	69.68
	CUBIC-BBR	50.38	287.47	219.90	39.09
	CUBIC-CUBIC	34.15	337.10	227.24	52.62
	Reno-BBR	58.93	288.15	225.43	39.19
	Reno-CUBIC	9.03	286.92	205.06	55.96

Table 6. Summary of packet loss statistics for the symmetric eMBB scenario.

Protocol	CC	Min	Max	μ	σ
MPTCP	BBR-BBR	0.000	0.203	0.030	0.032
	BBR-CUBIC	0.000	4.538	1.825	0.879
	CUBIC-BBR	0.000	0.612	0.162	0.114
	CUBIC-CUBIC	0.000	4.838	3.150	0.697
	Reno-BBR	0.000	0.926	0.260	0.194
	Reno-CUBIC	0.000	5.218	3.766	1.036
MPQUIC	BBR-BBR	0.000	0.000	0.000	0.000
	BBR-CUBIC	0.000	0.000	0.000	0.000
	CUBIC-BBR	0.000	0.000	0.000	0.000
	CUBIC-CUBIC	0.000	0.046	0.018	0.012
	Reno-BBR	0.000	0.579	0.104	0.112
	Reno-CUBIC	0.000	0.000	0.000	0.000

4.2. Symmetric Dual-Backhaul: mMTC-Inspired Scenario

We then consider the same symmetric dual-backhaul configuration under a denser mMTC-inspired traffic regime. In this case, 12,000 UEs generate low-rate traffic at 0.1 Mbps per device, again evenly split between MPTCP and MPQUIC sources. This setting is used

to stress contention and aggregation effects under balanced backhaul conditions, while highlighting the extent to which the considered CC pairings affect throughput sharing and jitter stability when a large number of low-rate flows concurrently access the same dual-backhaul infrastructure. Across the two selected CC pairings, the symmetric mMTC-inspired setting confirms that the balanced backhaul remains largely symmetric at link level, while the main differences emerge from how the two protocol stacks react to heavy contention and aggregation under dense low-rate traffic.

Under the CUBIC-BBR pairing (Figure 9), the cross-protocol imbalance is more evident than in the corresponding eMBB scenario. In Figure 9a the throughput distribution for the CUBIC-BBR pairing shows that MPTCP achieves higher throughput than MPQUIC on both terrestrial backhaul links, and the separation becomes more pronounced on TN2. In particular, MPTCP operates around $\mu = 248.67$ Mbps on TN1 and $\mu = 338.55$ Mbps on TN2, whereas MPQUIC reaches $\mu = 204.73$ Mbps on TN1 and drops to $\mu = 134.44$ Mbps on TN2. This indicates that, under heavy contention, the TCP-based stack retains a clearer advantage on both paths, while the QUIC-based stack is more penalized on the second backhaul link. Figure 9b reports the protocol independent jitter comparison between the two backhaul links. The two distributions largely overlap and remain concentrated at low values, with TN1 only slightly above TN2 in terms of mean jitter ($\mu = 1.919$ ms versus $\mu = 1.841$ ms), indicating comparable link-level delay variability. Figure 9c,d report protocol-specific jitter on TN1 and TN2, respectively. On TN1, MPTCP and MPQUIC exhibit similar mean jitter values, with $\mu = 4.265$ ms for MPTCP and $\mu = 4.414$ ms for MPQUIC. However, the MPQUIC distribution is much more dispersed and heavy-tailed. A similar behavior is observed on TN2, where MPQUIC has a lower mean jitter than MPTCP ($\mu = 3.694$ ms versus $\mu = 4.415$ ms) but still exhibits a markedly broader distribution. Therefore, under the CUBIC-BBR pairing, the main protocol-level difference in the massive-user setting lies less in the average jitter itself and more in the substantially larger variability of MPQUIC.

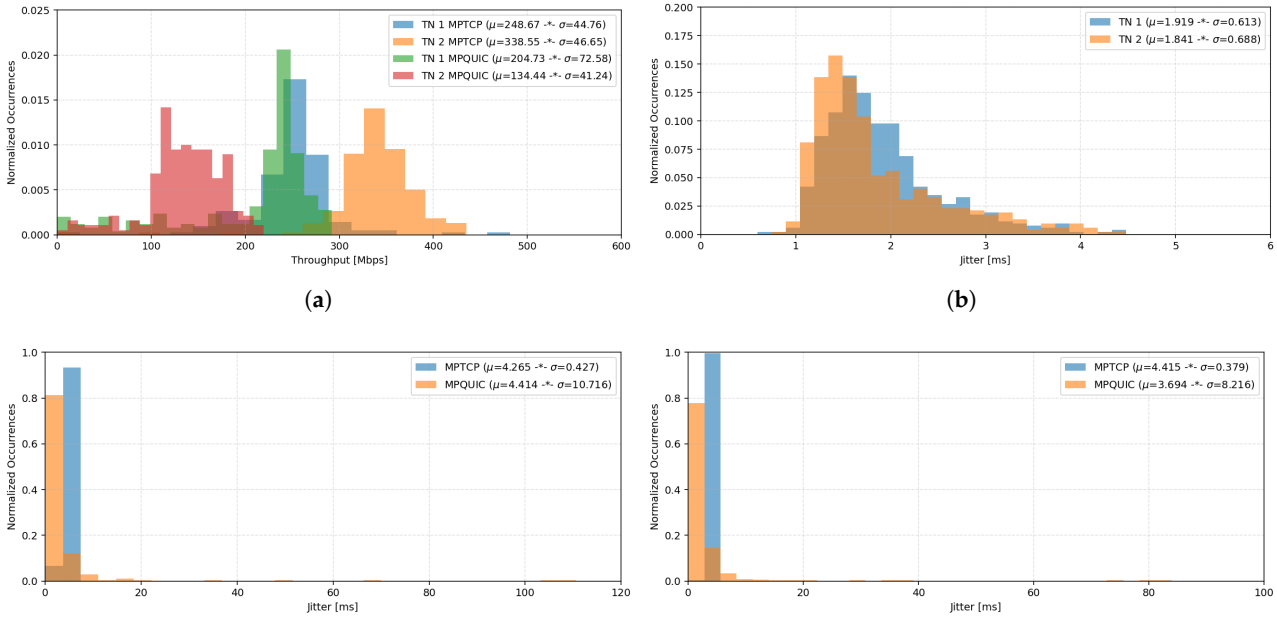

Figure 9. MPTCP and MPQUIC throughput and jitter distributions in symmetric mMTC-inspired scenario under the CUBIC-BBR CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

The Reno-BBR pairing (Figure 10a) preserves the same qualitative trend, with MPTCP outperforming MPQUIC on both terrestrial backhaul links. In particular, MPTCP reaches $\mu = 260.89$ Mbps on TN1 and $\mu = 321.82$ Mbps on TN2, while MPQUIC operates at $\mu = 193.13$ Mbps on TN1 and $\mu = 147.01$ Mbps on TN2. Hence, also under the Reno-BBR pairing, the throughput gap is moderate on TN1 and becomes more pronounced on TN2, confirming that the second backhaul link is the one where MPQUIC is more severely penalized under heavy load. Figure 10b reports the protocol independent jitter comparison across the two links. The TN1 and TN2 distributions again largely overlap, with very similar mean values ($\mu = 1.878$ ms on TN1 and $\mu = 1.775$ ms on TN2), indicating low and well-controlled link-level delay variability. However, Figure 10c,d reveal a clear protocol-level differentiation. On TN1, MPQUIC shows a slightly lower mean jitter than MPTCP ($\mu = 3.868$ ms versus $\mu = 4.246$ ms), but its distribution is much broader and more heavy-tailed. The same trend is observed on TN2, where MPQUIC remains lower on average ($\mu = 3.632$ ms) than MPTCP ($\mu = 4.330$ ms), while still exhibiting significantly larger dispersion. Therefore, under the Reno-BBR pairing, the two terrestrial links remain very similar at link level, whereas the main difference between the two protocol stacks is again the substantially higher jitter variability of MPQUIC despite its lower average operating point.

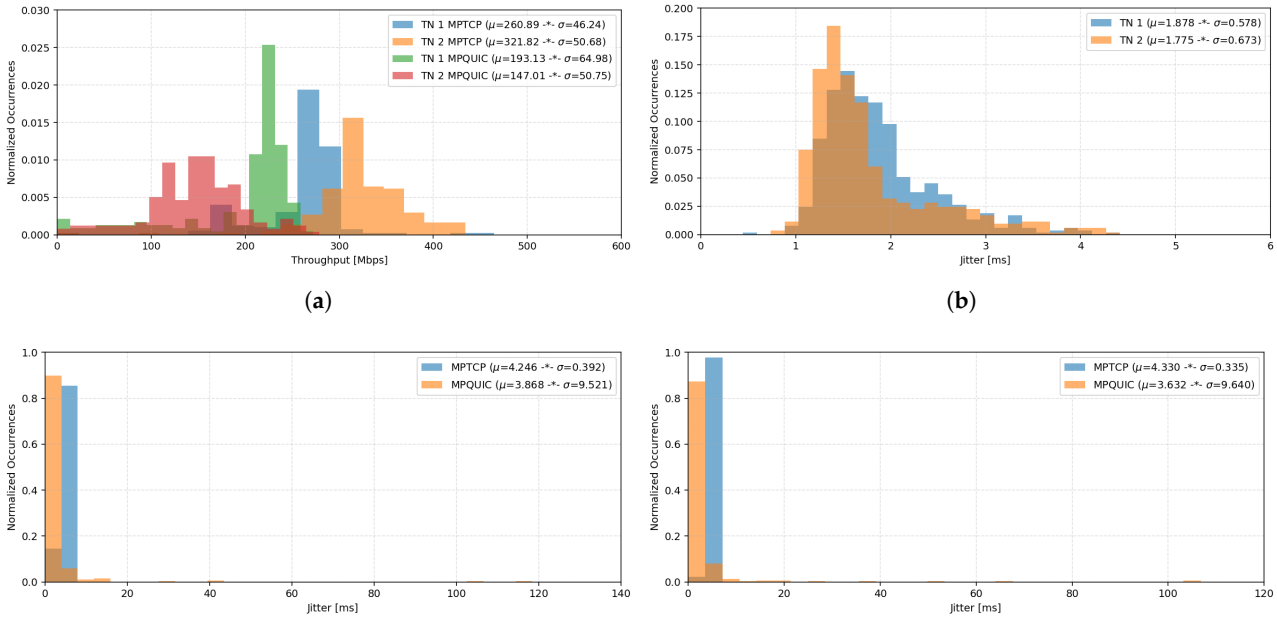

Figure 10. MPTCP and MPQUIC throughput and jitter distributions symmetric mMTC-inspired scenario under the Reno-BBR CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

Table 7 summarizes the mean jitter values observed for the two CC pairings considered in the symmetric mMTC-inspired scenario.

Table 7. Summary of the mean jitter values for the symmetric mMTC-inspired scenario.

MPTCP/MPQUIC CC	TN1	TN2	TN1 MPTCP	TN1 MPQUIC	TN2 MPTCP	TN2 MPQUIC
CUBIC-BBR	1.919	1.841	4.265	4.414	4.415	3.694
Reno-BBR	**1.878**	**1.775**	**4.246**	**3.868**	**4.330**	**3.632**

Therefore, the comparison between Figures 9 and 10, and Table 7 highlights a clear trade-off between throughput efficiency and jitter performance in the symmetric mMTC-inspired setting. From a throughput perspective, CUBIC-BBR is slightly more favorable, since it provides the highest aggregate mean throughput across the two terrestrial backhaul links, mainly due to the stronger MPTCP exploitation of TN2. From a jitter-oriented perspective, however, Reno-BBR is the better solution, as it achieves lower mean jitter both at link level and at protocol-specific level on TN1 and TN2. Hence, also under heavy contention, the pairing that maximizes throughput is not necessarily the one that minimizes delay variability. CUBIC-BBR is therefore preferable when throughput is the primary objective, whereas Reno-BBR is the more suitable option when lower and more stable jitter is the main design target.

Tables 8 and 9 show that CUBIC-BBR provides the highest overall throughput in the symmetric mMTC-inspired setting, mainly because of the stronger MPTCP exploitation of TN2. However, this does not automatically make it the most balanced choice in general terms, since the subsection highlights a trade-off with Reno-BBR, which is preferable from the jitter standpoint. Therefore, also here, the throughput-leading configuration should be interpreted as a throughput-oriented choice rather than as a universally best one.

Table 8. Summary of MPTCP throughput statistics for the symmetric mMTC-inspired scenario.

Path	CC	Min	Max	μ	σ
TN1	CUBIC-BBR	136.11	481.55	248.67	44.76
	Reno-BBR	159.58	464.56	260.89	46.24
TN2	CUBIC-BBR	103.38	435.17	338.55	46.65
	Reno-BBR	88.83	434.01	321.82	50.68

Table 9. Summary of MPQUIC throughput statistics for the symmetric mMTC-inspired scenario.

Path	CC	Min	Max	μ	σ
TN1	CUBIC-BBR	3.16	291.98	204.73	72.58
	Reno-BBR	2.83	272.34	193.13	64.98
TN2	CUBIC-BBR	12.87	219.74	134.44	41.24
	Reno-BBR	12.96	278.95	147.01	50.75

Table 10 shows that, under dense mMTC-inspired traffic, packet loss remains systematically higher for MPQUIC than for MPTCP under both considered pairings, consistent with the throughput results, which showed that the TCP-based stack preserves a clearer advantage under heavy contention. Between the two configurations, CUBIC-BBR provides lower MPQUIC packet loss than Reno-BBR, while the MPTCP loss level remains comparable across the two cases. This observation is coherent with the throughput discussion, where CUBIC-BBR was found to be slightly preferable when throughput efficiency is the main objective. At the same time, Reno-BBR was previously identified as the better jitter-oriented solution; therefore, Table 10 suggests that Reno-BBR should be interpreted as a delay-stability-oriented choice rather than a loss-minimizing one. In contrast, CUBIC-BBR emerges as a more balanced compromise in the symmetric mMTC-inspired scenario, since it combines the highest aggregate throughput with somewhat lower packet-loss impact, especially on the MPQUIC side.

Table 10. Summary of packet loss statistics for the symmetric mMTC-inspired scenario.

Protocol	CC	Min	Max	μ	σ
MPTCP	CUBIC-BBR	0.000	2.617	0.462	0.262
	Reno-BBR	0.000	2.788	0.436	0.262
MPQUIC	CUBIC-BBR	0.000	2.942	1.523	1.245
	Reno-BBR	0.000	4.349	2.299	1.806

4.3. Asymmetric Dual-Backhaul: eMBB Scenario

We next move to the asymmetric dual-backhaul setting, where the two terrestrial backhaul links have different lengths, namely 10 km for TN1 and 15 km for TN2. As in the symmetric eMBB case, 12 UEs generate traffic at 100 Mbps per device, with half of the users employing MPTCP and the other half MPQUIC. This scenario is used to evaluate how backhaul heterogeneity interacts with the selected CC pairings and affects throughput balance, fairness, and jitter under shared transport operation. Across the considered CC pairings, the asymmetric setting shows that the second terrestrial backhaul link becomes the main point where protocol-level differences are amplified, so that throughput imbalance and jitter variability are no longer shaped only by the selected CC combination, but also by the heterogeneous path conditions.

As shown in Figure 11a, under the BBR-BBR pairing MPTCP achieves consistently higher throughput than MPQUIC on both terrestrial backhaul links. In particular, MPTCP operates around $\mu = 323.98$ Mbps on TN1 and $\mu = 246.49$ Mbps on TN2, while MPQUIC attains lower mean rates of $\mu = 172.48$ Mbps on TN1 and $\mu = 140.97$ Mbps on TN2, indicating a persistent throughput advantage for the TCP-based stack even when both protocols adopt BBR. Figure 11b reports the protocol independent jitter distributions for the two backhaul links, which largely overlap, with TN2 only slightly shifted toward higher values ($\mu = 1.614$ ms versus $\mu = 1.511$ ms on TN1), suggesting comparable link-level delay variability. Figure 11c,d isolate protocol-specific jitter on TN1 and TN2, respectively. On both links, MPQUIC remains centered at lower jitter values ($\mu = 2.074$ ms on TN1 and $\mu = 2.083$ ms on TN2), whereas MPTCP is shifted to higher operating regions ($\mu = 3.647$ ms on TN1 and $\mu = 4.999$ ms on TN2). Overall, with BBR on both stacks, the throughput asymmetry remains pronounced in favor of MPTCP, while the two terrestrial links exhibit similarly low link-level jitter and a stable protocol-level separation.

In Figure 12a, the throughput distributions for the BBR-CUBIC pairing show that MPQUIC now attains the highest operating regions on both terrestrial backhaul links. In particular, MPQUIC reaches $\mu = 270.33$ Mbps on TN1 and $\mu = 229.52$ Mbps on TN2, while MPTCP remains lower at $\mu = 226.90$ Mbps on TN1 and $\mu = 157.86$ Mbps on TN2. This indicates a reversal of the throughput balance observed in the BBR-BBR case, with the QUIC-based stack becoming dominant under this pairing. Figure 12b reports the protocol independent jitter comparison between TN1 and TN2. The TN2 distribution is shifted to higher values and is visibly broader, with $\mu = 3.547$ ms versus $\mu = 2.717$ ms on TN1, indicating a more variable timing behavior on the second backhaul link. Figure 12c,d reveal that MPQUIC also exhibits higher protocol-specific jitter than MPTCP on both links: on TN1, MPQUIC reaches $\mu = 6.910$ ms versus $\mu = 4.173$ ms for MPTCP, while on TN2 the gap remains evident ($\mu = 7.634$ ms versus $\mu = 5.725$ ms). Therefore, under the BBR-CUBIC pairing, MPQUIC gains a throughput advantage, but this comes at the cost of significantly higher inter-arrival variability on both terrestrial links.

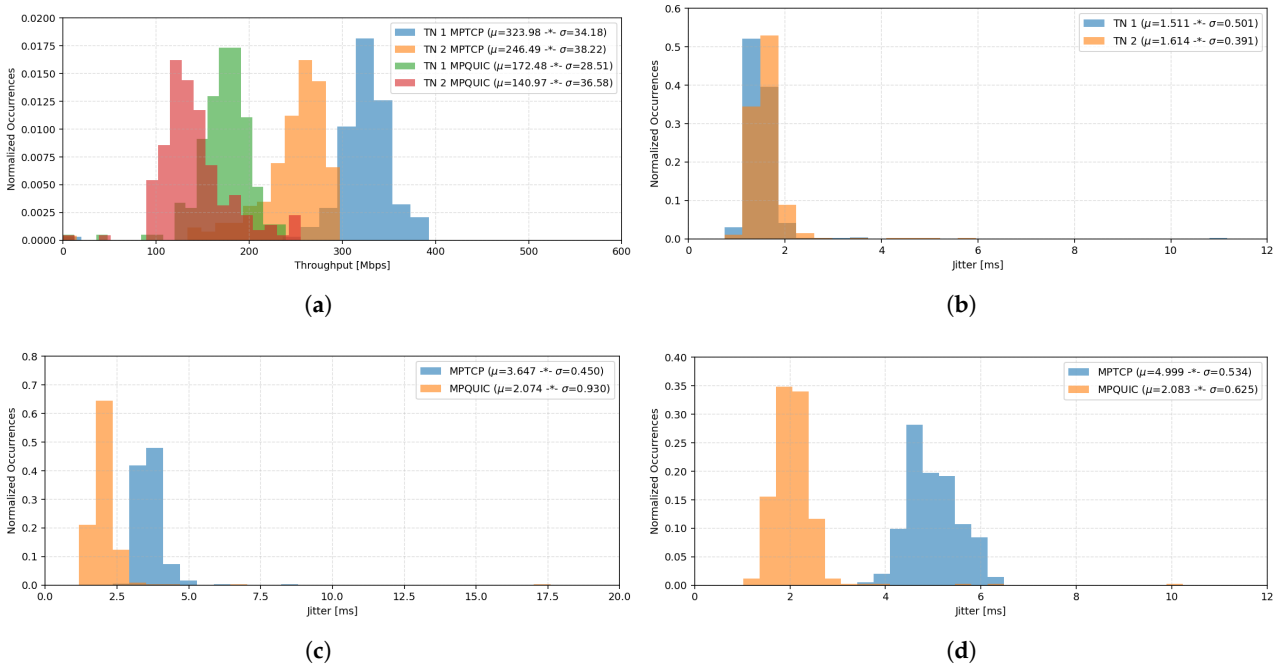

Figure 11. MPTCP and MPQUIC throughput and jitter distributions in asymmetric eMBB scenario under the BBR-BBR CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

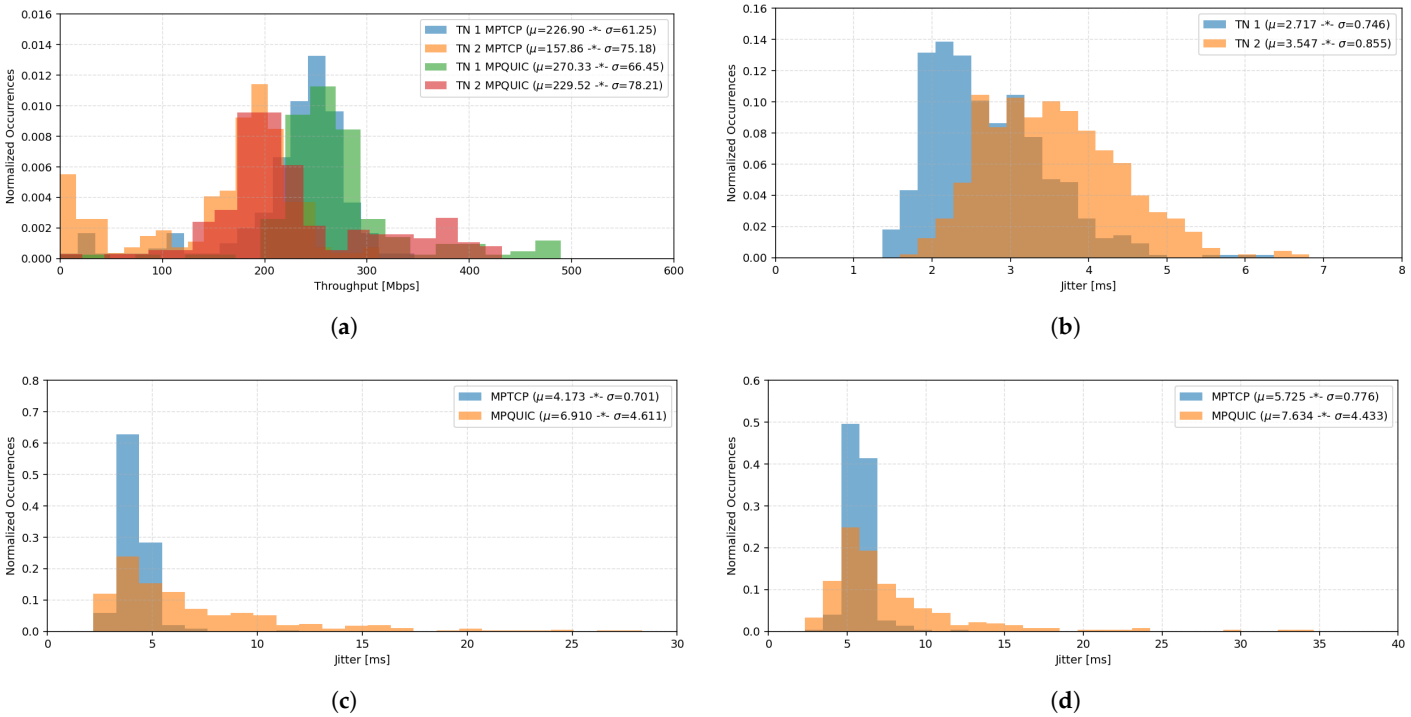

Figure 12. MPTCP and MPQUIC throughput and jitter distributions in asymmetric eMBB scenario under the BBR-CUBIC CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

In Figure 13a, the throughput distributions for the CUBIC-BBR pairing show a strong and consistent advantage for MPTCP over MPQUIC on both terrestrial backhaul links. MPTCP reaches $\mu = 333.25$ Mbps on TN1 and $\mu = 259.36$ Mbps on TN2 while MPQUIC

remains substantially lower, with $\mu = 163.17$ Mbps on TN1 and $\mu = 128.09$ Mbps on TN2. The two protocol stacks therefore occupy clearly separated throughput regions, indicating that this pairing strongly favors the TCP-based implementation. Figure 13b reports the protocol independent jitter comparison, showing tightly overlapping distributions with very low averages on both links ($\mu = 1.490$ ms on TN1 and $\mu = 1.636$ ms on TN2), which indicates limited link-level delay variability. Figure 13c,d isolate protocol-specific jitter on TN1 and TN2, respectively. On both links, MPQUIC exhibits lower average jitter than MPTCP ($\mu = 2.106$ ms vs. $\mu = 3.773$ ms on TN1 and $\mu = 2.114$ ms vs. $\mu = 5.335$ ms on TN2), although its distributions remain somewhat broader. Overall, this pairing combines a clear throughput dominance of MPTCP with the lowest link-level jitter among the considered configurations, while still preserving comparatively low protocol-specific jitter for MPQUIC.

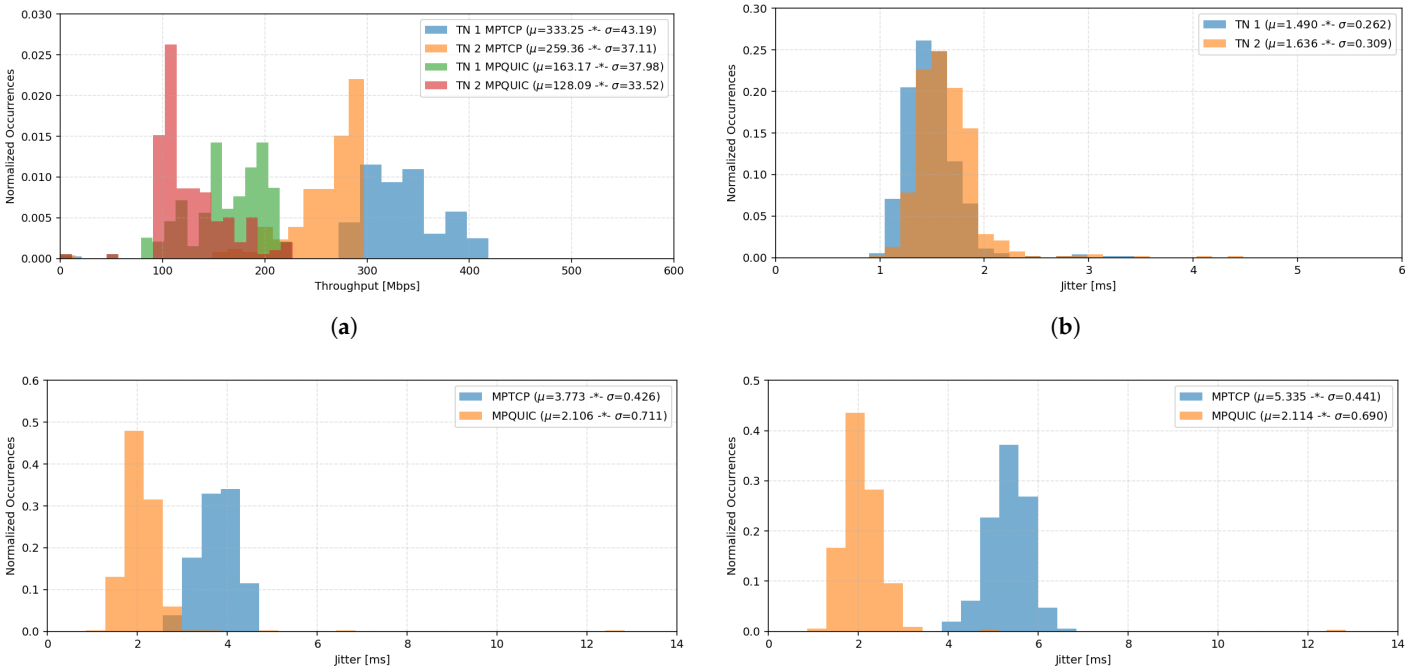

Figure 13. MPTCP and MPQUIC throughput and jitter distributions in asymmetric eMBB scenario under the CUBIC-BBR CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

In Figure 14a, the CUBIC-CUBIC pairing again favors MPTCP on both terrestrial backhaul links, although the throughput gap is less extreme than in the previous case on TN1. MPTCP operates around $\mu = 304.04$ Mbps on TN1 and $\mu = 264.79$ Mbps on TN2, while MPQUIC reaches $\mu = 193.73$ Mbps on TN1 and drops to $\mu = 122.37$ Mbps on TN2, indicating a stronger imbalance on the second backhaul link. Figure 14b reports the protocol independent jitter comparison and shows that TN2 is shifted to higher values and exhibits a broader spread than TN1, with means of $\mu = 3.500$ ms and $\mu = 2.667$ ms, respectively. Figure 14c,d reveal a clear protocol-level differentiation: on TN1, MPQUIC already exhibits higher and more dispersed jitter than MPTCP ($\mu = 5.734$ ms versus $\mu = 3.959$ ms), and the difference becomes more pronounced on TN2, where MPQUIC reaches $\mu = 8.874$ ms while MPTCP remains centered around $\mu = 5.058$ ms. Hence, under the CUBIC-CUBIC pairing, MPTCP retains a throughput advantage, while MPQUIC experiences noticeably larger timing variability, especially on the second terrestrial link.

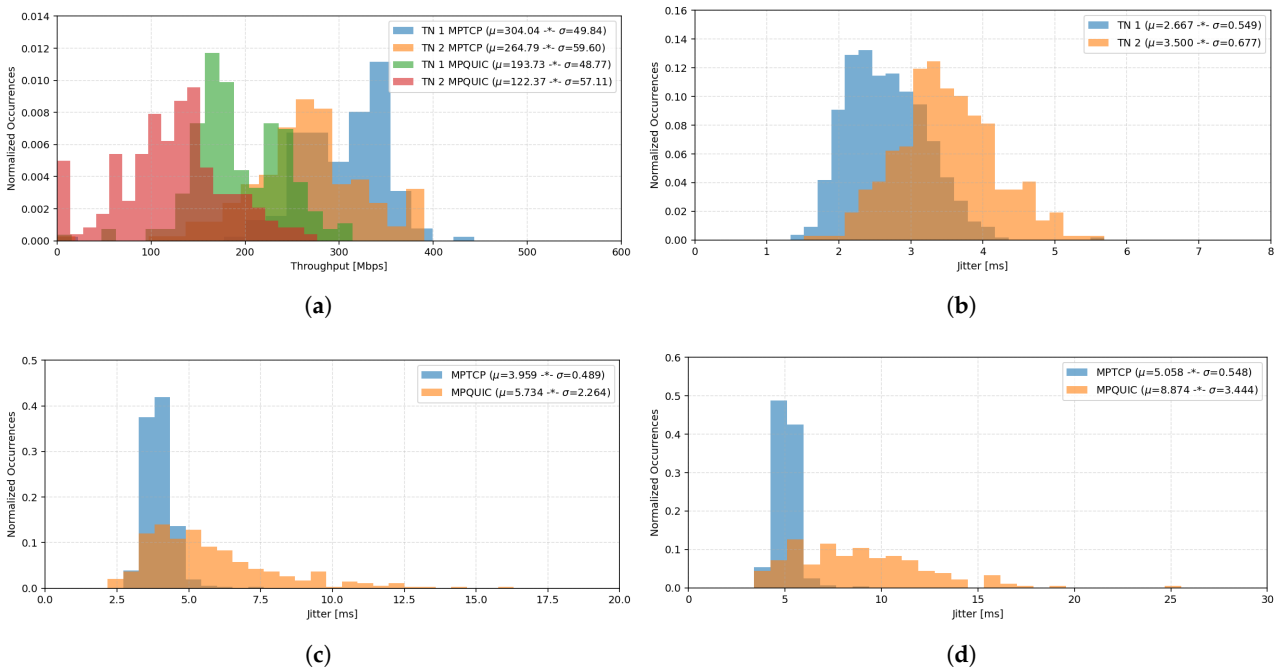

Figure 14. MPTCP and MPQUIC throughput and jitter distributions in asymmetric eMBB scenario under the CUBIC-CUBIC CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

In Figure 15a, the throughput distributions for the Reno-BBR pairing show a consistent advantage of MPTCP over MPQUIC on both terrestrial backhaul links. In particular, MPTCP achieves mean rates of $\mu = 323.27$ Mbps on TN1 and $\mu = 259.40$ Mbps on TN2, while MPQUIC remains substantially lower with $\mu = 172.94$ Mbps on TN1 and $\mu = 128.09$ Mbps on TN2. This indicates a stable cross-protocol imbalance in favor of the TCP-based stack, with both links showing a similar ordering. Figure 15b compares protocol independent jitter across TN1 and TN2 and shows largely overlapping distributions, with TN2 only slightly shifted to higher values ($\mu = 1.594$ ms versus $\mu = 1.492$ ms on TN1). Figure 15c,d isolate protocol-specific jitter on TN1 and TN2, respectively. On both links, MPQUIC remains centered at lower jitter values ($\mu = 2.055$ ms on TN1 and $\mu = 2.056$ ms on TN2), whereas MPTCP operates at higher jitter levels ($\mu = 3.678$ ms on TN1 and $\mu = 5.104$ ms on TN2) with a more compact core distribution. Therefore, under Reno-BBR, MPTCP dominates throughput on both links, while MPQUIC preserves lower average jitter despite a somewhat broader dispersion.

In Figure 16a, the throughput distributions for the Reno-CUBIC pairing show a more link-dependent cross-protocol balance. On TN1, MPTCP and MPQUIC operate in relatively close throughput regions, with MPTCP attaining $\mu = 274.48$ Mbps and MPQUIC $\mu = 221.89$ Mbps. On TN2, however, the separation becomes much stronger: MPTCP remains high at $\mu = 275.53$ Mbps, whereas MPQUIC drops to $\mu = 111.99$ Mbps, indicating a marked reduction of QUIC throughput on the second terrestrial backhaul link. Figure 16b reports protocol-independent jitter at the link level and shows that TN2 is shifted toward higher values and broader variability, with $\mu = 3.782$ ms compared with $\mu = 3.023$ ms on TN1. Figure 16c,d separate protocol-specific jitter on TN1 and TN2, respectively. On TN1, MPQUIC already exhibits higher and more dispersed jitter than MPTCP ($\mu = 8.224$ ms versus $\mu = 4.258$ ms), and this difference becomes much more pronounced on TN2, where MPQUIC reaches $\mu = 11.251$ ms while MPTCP remains centered around $\mu = 5.258$ ms. Hence, Reno-CUBIC combines a moderate throughput imbalance on TN1 with a severe

degradation of MPQUIC on TN2, and it also induces the largest protocol-level jitter penalty for MPQUIC among the considered asymmetric-link configurations.

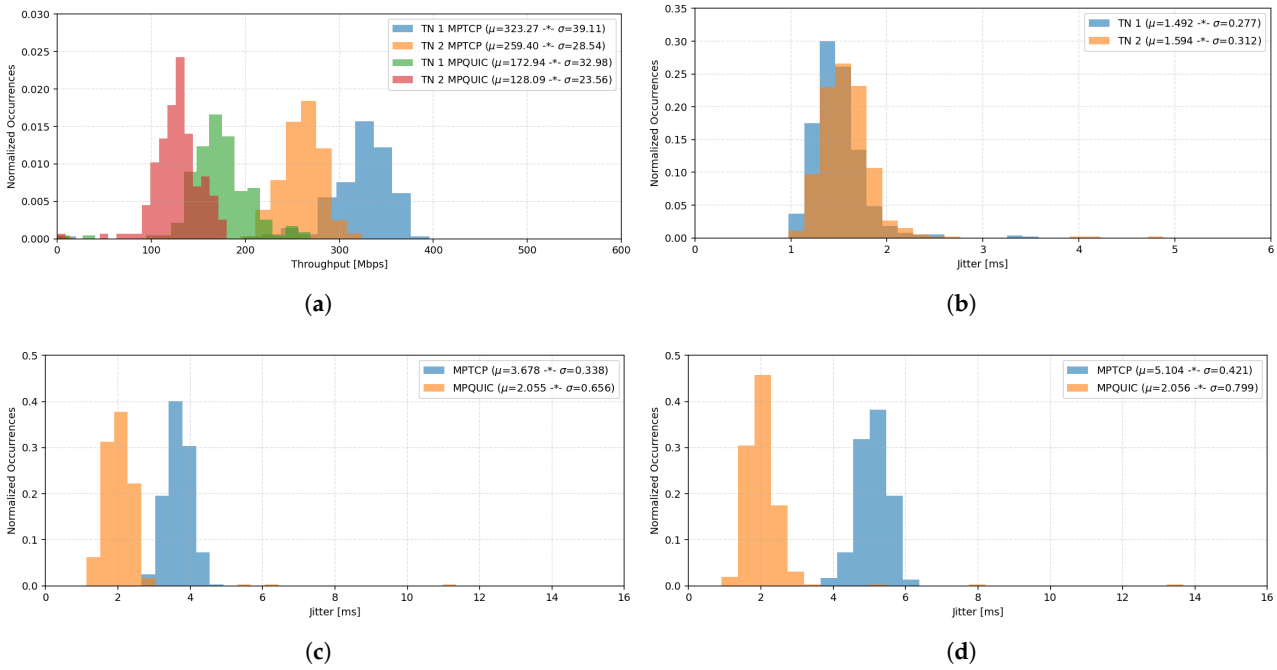

Figure 15. MPTCP and MPQUIC throughput and jitter distributions in asymmetric eMBB scenario under the Reno-BBR CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

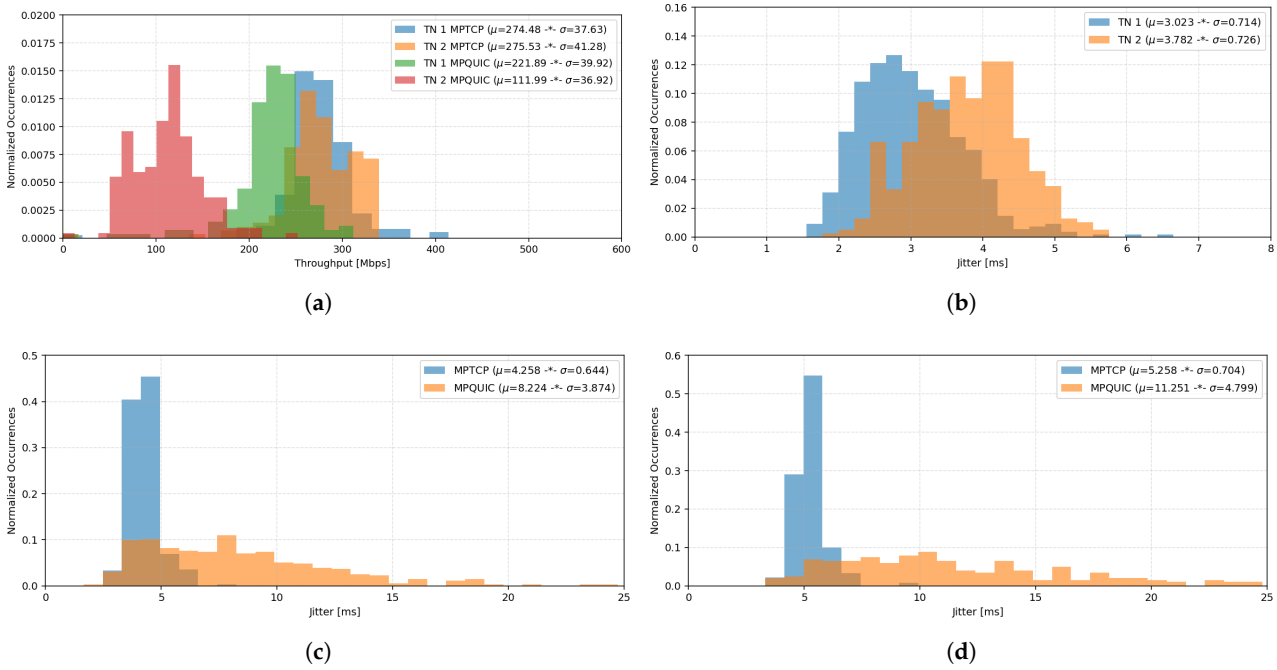

Figure 16. MPTCP and MPQUIC throughput and jitter distributions in asymmetric eMBB scenario under the Reno-CUBIC CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

Beyond the per-pairing distributions, Tables 11 and 12 summarize the fairness and mean jitter trends observed across the considered CC combinations in the asymmetric setting.

Table 11. Jain's fairness index between MPTCP and MPQUIC in asymmetric eMBB scenario computed from the mean throughput values reported in panels (a) of the figures from Figures 11–16.

MPTCP/MPQUIC CC	TN1	TN2	Total
BBR-BBR	0.9148	0.9310	0.9220
BBR-CUBIC	**0.9924**	**0.9669**	**0.9834**
CUBIC-BBR	0.8949	0.8970	0.8959
CUBIC-CUBIC	0.9532	0.8808	0.9246
Reno-BBR	0.9159	0.8970	0.9078
Reno-CUBIC	0.9889	0.8488	0.9436

Table 12. Summary of the mean jitter values extracted from Figures 11–16 for the asymmetric dual-link backhaul eMBB scenario. Boldface highlights the CC pairings providing the most favorable fairness performance.

MPTCP/MPQUIC CC	TN1	TN2	TN1 MPTCP	TN1 MPQUIC	TN2 MPTCP	TN2 MPQUIC
BBR-BBR	**1.511**	**1.614**	**3.647**	**2.074**	**4.999**	**2.083**
BBR-CUBIC	2.717	3.547	4.173	6.910	5.725	7.634
CUBIC-BBR	1.490	1.636	3.773	2.106	5.335	2.114
CUBIC-CUBIC	2.667	3.500	3.959	5.734	5.058	8.874
Reno-BBR	**1.492**	**1.594**	**3.678**	**2.055**	**5.104**	**2.056**
Reno-CUBIC	3.023	3.782	4.258	8.224	5.258	11.251

The fairness results in Table 11 show a clearer trend than in the symmetric scenario. In this case, BBR-CUBIC emerges as the single best CC pairing from the fairness standpoint, since it achieves the highest Jain's index on both terrestrial backhaul links as well as the highest aggregate fairness. Therefore, unlike the symmetric case, where two pairings could be highlighted depending on whether fairness was evaluated per-link or over the aggregate throughput, here the same configuration dominates under both criteria. If a secondary reference is to be identified, Reno-CUBIC provides the second-highest aggregate fairness and remains very close to BBR-CUBIC on TN1. However, its much lower fairness on TN2 prevents it from being regarded as equally effective from a per-link fairness perspective.

From a jitter-oriented perspective, Table 12 indicates that the most favorable CC pairings in the asymmetric dual-link backhaul scenario are Reno-BBR and BBR-BBR. In particular, Reno-BBR provides the lowest mean jitter across TN1 and TN2, both at link level and at protocol-specific level, while BBR-BBR exhibits very similar performance and confirms the same overall trend. By contrast, pairings such as BBR-CUBIC and Reno-CUBIC induce substantially larger jitter dispersion, especially for MPQUIC, whereas CUBIC-CUBIC also leads to higher protocol-level jitter on the more penalized backhaul link. For this reason, the following mMTC-inspired analysis focuses on Reno-BBR and BBR-BBR, as the two configurations that emerge as the most representative jitter-efficient solutions in the asymmetric scenario.

Tables 13 and 14 further confirm that maximum throughput and fairness do not necessarily coincide. In fact, the highest overall throughput is obtained only slightly by CUBIC-CUBIC, whereas Table 11 identifies BBR-CUBIC as the clearest fairness-optimal solution on both links and on the aggregate throughput. Thus, in the asymmetric eMBB case, the throughput-maximizing pairing is again different from the fairness-maximizing one.

Table 13. Summary of MPTCP throughput statistics for the asymmetric eMBB scenario.

Path	CC	Min	Max	μ	σ
TN1	BBR-BBR	252.66	392.77	323.98	23.77
	BBR-CUBIC	23.24	346.22	226.90	58.94
	CUBIC-BBR	278.88	418.46	333.25	35.07
	CUBIC-CUBIC	188.31	443.43	304.04	44.28
	Reno-BBR	215.22	395.99	323.27	30.52
	Reno-CUBIC	186.57	414.21	274.48	31.40
TN2	BBR-BBR	135.94	297.45	246.49	33.41
	BBR-CUBIC	0.00	311.92	157.86	74.43
	CUBIC-BBR	161.91	297.13	259.36	31.53
	CUBIC-CUBIC	116.31	390.48	264.79	56.26
	Reno-BBR	210.04	324.19	259.40	20.69
	Reno-CUBIC	144.86	339.59	275.53	35.67

Table 14. Summary of MPQUIC throughput statistics for the asymmetric eMBB scenario.

Path	CC	Min	Max	μ	σ
TN1	BBR-BBR	43.96	239.03	172.48	25.39
	BBR-CUBIC	41.22	489.31	270.33	63.37
	CUBIC-BBR	52.66	225.97	163.17	36.00
	CUBIC-CUBIC	48.12	313.92	193.73	46.62
	Reno-BBR	39.25	269.48	172.94	30.33
	Reno-CUBIC	51.08	311.84	221.89	36.28
TN2	BBR-BBR	50.84	255.08	140.97	35.08
	BBR-CUBIC	49.26	432.22	229.52	76.46
	CUBIC-BBR	52.71	227.29	128.09	32.16
	CUBIC-CUBIC	3.16	276.46	122.37	56.51
	Reno-BBR	53.48	180.28	128.09	21.52
	Reno-CUBIC	48.63	251.73	111.99	36.03

Table 15 confirms that packet-loss behavior in the asymmetric eMBB scenario is broadly consistent with the throughput trends observed in the corresponding figures and summary tables. In particular, pairings that strongly favor one protocol over the other in throughput terms also tend to exhibit more pronounced loss asymmetries, especially on the MPTCP side. From this perspective, BBR-BBR, CUBIC-BBR, and Reno-BBR remain comparatively well behaved, with relatively limited loss values, whereas configurations such as BBR-CUBIC, CUBIC-CUBIC, and especially Reno-CUBIC show a substantially less favorable loss profile. This does not invalidate the fairness-oriented role of BBR-CUBIC, which remains the most favorable configuration from the fairness standpoint in the asymmetric scenario, but it indicates that such fairness gains are achieved at the price of a less favorable loss behavior. By contrast, Reno-BBR and BBR-BBR, already highlighted in the jitter analysis as the most representative jitter-efficient solutions, also preserve a more contained packet-loss profile and can therefore be regarded as stronger overall compromises when robustness and delay stability are prioritized over fairness alone.

4.4. Asymmetric Dual-Backhaul: mMTC-Inspired Scenario

The asymmetric setting is then evaluated under the mMTC-inspired traffic regime, where 12,000 UEs generate low-rate traffic at 0.1 Mbps per device, again evenly partitioned between MPTCP-enabled and MPQUIC-enabled sources. In this case, the analysis captures the combined effect of backhaul heterogeneity and dense low-rate traffic on transport coexistence, with particular attention to jitter behavior and throughput sharing across the

two protocol families. Across the two selected CC pairings, the asymmetric mMTC-inspired setting confirms that the second terrestrial backhaul link is the most penalized operating point, where protocol-level differences become more pronounced under dense low-rate traffic and heterogeneous path conditions.

Table 15. Summary of packet loss statistics for the asymmetric eMBB scenario.

Path	CC	Min	Max	μ	σ
MPTCP	BBR-BBR	0.000	0.000	0.000	0.000
	BBR-CUBIC	0.000	2.008	1.019	0.354
	CUBIC-BBR	0.000	0.411	0.186	0.091
	CUBIC-CUBIC	0.000	1.916	1.282	0.603
	Reno-BBR	0.000	0.053	0.016	0.011
	Reno-CUBIC	0.000	3.033	1.522	0.635
MPQUIC	BBR-BBR	0.000	0.000	0.000	0.000
	BBR-CUBIC	0.000	0.000	0.000	0.000
	CUBIC-BBR	0.000	0.046	0.010	0.009
	CUBIC-CUBIC	0.000	0.027	0.004	0.009
	Reno-BBR	0.000	0.000	0.000	0.000
	Reno-CUBIC	0.000	0.020	0.003	0.003

Under the BBR-BBR pairing (Figure 17), the asymmetric mMTC-inspired setting reveals a marked link-dependent cross-protocol allocation. As shown in Figure 17a, on TN1, MPTCP and MPQUIC operate in relatively close throughput regions, with MPQUIC slightly higher ($\mu = 276.72$ Mbps) than MPTCP ($\mu = 257.26$ Mbps). On TN2, by contrast, the separation becomes much stronger: MPTCP remains at a high operating region ($\mu = 251.47$ Mbps), whereas MPQUIC drops to a much lower throughput level ($\mu = 85.97$ Mbps). This indicates that, under heavy contention, the QUIC-based stack is still able to exploit TN1 effectively, but loses most of its contribution on TN2, while MPTCP retains a more stable use of both backhaul links. Figure 17b reports the protocol independent jitter comparison between TN1 and TN2 and shows largely overlapping distributions, with only a slight increase on TN2 ($\mu = 2.354$ ms versus $\mu = 2.246$ ms on TN1), suggesting comparable link-level delay variability. Figure 17c,d isolate protocol-specific jitter on TN1 and TN2, respectively. On TN1, MPQUIC exhibits a slightly higher mean jitter than MPTCP ($\mu = 5.364$ ms versus $\mu = 4.211$ ms) together with a much broader and heavier-tailed distribution. On TN2, the average jitter values remain relatively close ($\mu = 5.004$ ms for MPQUIC and $\mu = 5.643$ ms for MPTCP), but MPQUIC again shows much larger dispersion and pronounced outliers. Overall, the BBR-BBR pairing in the massive asymmetric scenario combines a moderate throughput split on TN1 with a sharp degradation of MPQUIC on TN2, while the protocol-level jitter remains substantially less stable for MPQUIC on both links.

The Reno-BBR pairing (Figure 18) yields a similar but slightly more balanced behavior on TN1. As shown in Figure 18a, MPQUIC attains a marginally higher mean throughput than MPTCP on TN1 ($\mu = 298.89$ Mbps versus $\mu = 280.95$ Mbps), while in TN2 the cross-protocol separation becomes pronounced, with MPTCP operating around $\mu = 190.68$ Mbps and MPQUIC dropping to $\mu = 83.24$ Mbps. Hence, under Reno-BBR, the less favorable backhaul path is mainly exploited by MPTCP, while MPQUIC remains concentrated on TN1 and contributes much less on TN2. Figure 18b reports the protocol independent jitter comparison and again shows closely overlapping low-jitter distributions on the two links, with TN2 slightly shifted upward ($\mu = 2.123$ ms versus $\mu = 1.893$ ms on TN1). Figure 18c,d separate protocol-specific jitter on TN1 and TN2. On both links, MPQUIC attains slightly lower average jitter than MPTCP ($\mu = 3.861$ ms vs. $\mu = 4.026$ ms on TN1, and $\mu = 4.140$ ms

vs. $\mu = 5.564$ ms on TN2). Nevertheless, it is again characterized by much broader, heavy-tailed distributions, whereas MPTCP remains tightly concentrated around its operating region. Therefore, the Reno-BBR pairing preserves very low link-level jitter and a relatively balanced use of TN1. However, it still leads to a marked reduction of MPQUIC throughput on TN2 and significantly less stable protocol-level jitter for the QUIC-based stack.

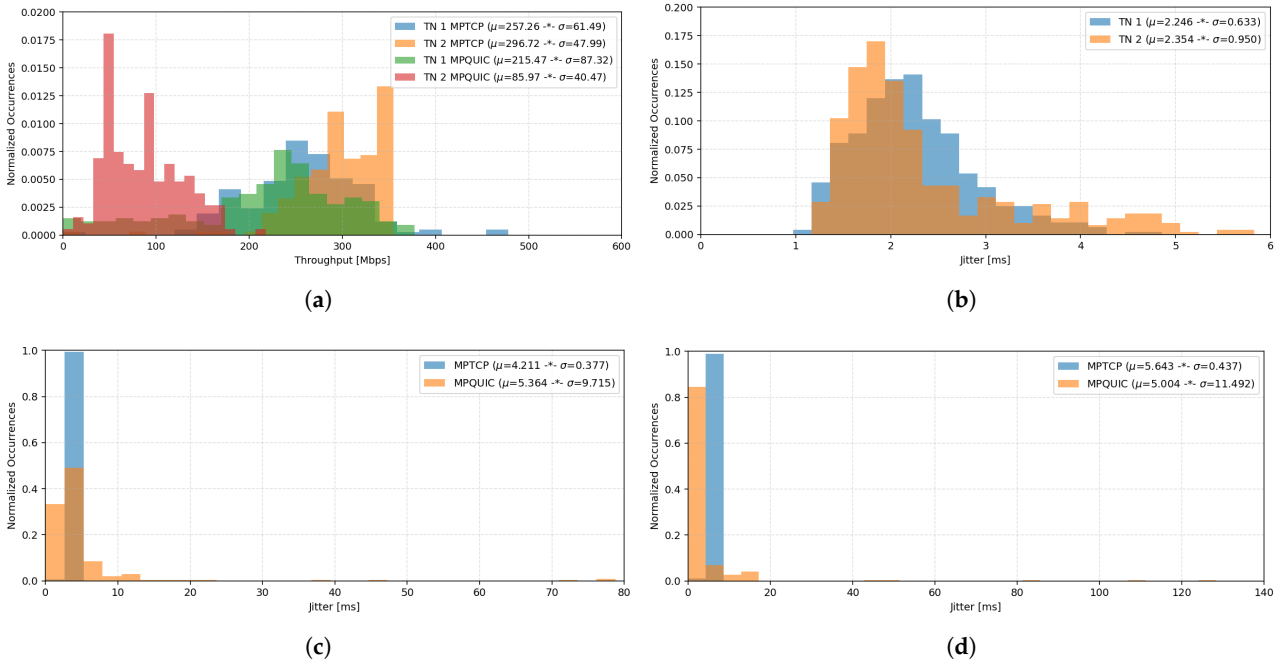

Figure 17. MPTCP and MPQUIC throughput and jitter distributions in asymmetric mMTC-inspired scenario under the BBR-BBR CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

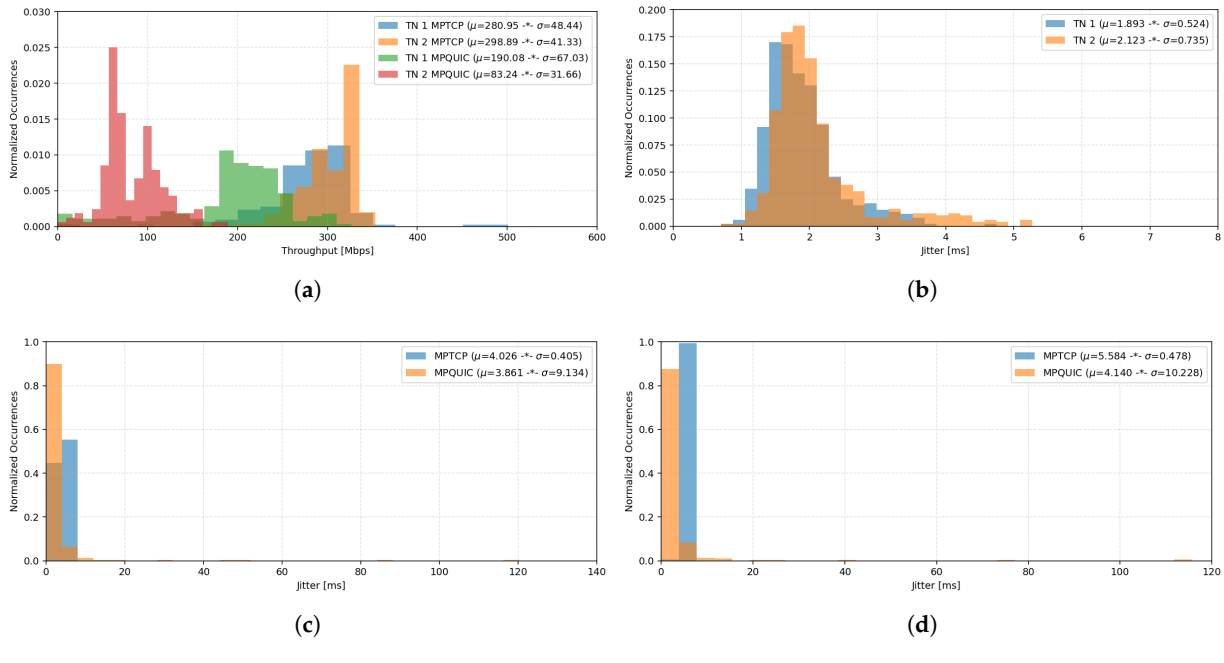

Figure 18. MPTCP and MPQUIC throughput and jitter distributions in asymmetric mMTC-inspired scenario under the Reno-BBR CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

Table 16 summarizes the mean jitter values observed for the two CC pairings considered in the asymmetric mMTC-inspired scenario.

Table 16. Summary of the mean jitter values for the asymmetric mMTC-inspired scenario.

MPTCP/MPQUIC CC	TN1	TN2	TN1 MPTCP	TN1 MPQUIC	TN2 MPTCP	TN2 MPQUIC
BBR-BBR	2.246	2.354	4.211	5.364	5.643	5.004
Reno-BBR	**1.893**	**2.123**	**4.026**	**3.861**	**5.564**	**4.140**

Therefore, the joint comparison of Figures 17 and 18, and Table 16 highlights a clear trade-off between throughput efficiency and jitter performance in the asymmetric mMTC-inspired scenario. From a throughput perspective, BBR-BBR is slightly more favorable, since it enables a stronger overall exploitation of the two terrestrial backhaul links, mainly due to the higher MPTCP operating region, especially on TN2. From a jitter-oriented perspective, however, Reno-BBR represents the better solution, as it provides lower and more stable jitter both at link level and at protocol-specific level on TN1 and TN2. Hence, also in the asymmetric mMTC-inspired setting, the pairing that maximizes throughput is not necessarily the one that minimizes delay variability. BBR-BBR is therefore preferable when throughput is the main objective, whereas Reno-BBR is the most suitable option when lower and more stable jitter is the primary design target.

Tables 17 and 18 show that BBR-BBR is slightly more favorable from the throughput perspective in the asymmetric mMTC-inspired regime, mainly due to the stronger MPTCP operating region, especially on TN2. Nevertheless, the same subsection indicates Reno-BBR as the more suitable solution when the target shifts toward stability. Hence, the pairing that maximizes throughput is not the one that emerges as the most balanced overall choice.

Table 17. Summary of MPTCP throughput statistics for the asymmetric mMTC-inspired scenario.

Path	CC	Min	Max	μ	σ
TN1	BBR-BBR	126.42	478.35	257.26	61.49
	Reno-BBR	151.54	500.84	280.95	48.44
TN2	BBR-BBR	87.28	355.03	296.72	47.99
	Reno-BBR	100.83	353.26	298.89	41.33

Table 18. Summary of MPQUIC throughput statistics for the asymmetric mMTC-inspired scenario.

Path	CC	Min	Max	μ	σ
TN1	BBR-BBR	3.18	377.68	215.47	87.32
	Reno-BBR	3.18	326.69	190.08	67.03
TN2	BBR-BBR	13.70	217.50	85.97	40.47
	Reno-BBR	13.30	189.69	83.24	31.66

Table 19 provides an additional perspective on the trade-off already highlighted in the asymmetric mMTC-inspired scenario. The packet-loss results show that Reno-BBR yields lower average loss than BBR-BBR on both protocol stacks, most notably on the MPQUIC side, where the reduction is clearly visible. This is coherent with the previous jitter-oriented analysis, which had already identified Reno-BBR as the most suitable solution when lower and more stable delay is the primary design target. On the other hand, BBR-BBR had been found slightly more favorable from the throughput perspective, mainly because of its stronger MPTCP exploitation of the two backhaul links, especially TN2. Therefore, Table 19 reinforces the same interpretation rather than changing it: BBR-BBR remains the preferable

option when throughput efficiency is the main objective, whereas Reno-BBR becomes the more balanced solution when the design focus also includes loss containment in addition to jitter stability. In this sense, the packet-loss statistics further strengthen Reno-BBR as the safer operating point from a robustness perspective, while BBR-BBR remains attractive as a throughput-oriented choice.

Table 19. Summary of packet loss statistics for the asymmetric mMTC-inspired scenario.

Path	CC	Min	Max	μ	σ
MPTCP	BBR-BBR	0.000	1.478	0.186	0.151
	Reno-BBR	0.000	1.676	0.150	0.187
MPQUIC	BBR-BBR	0.000	4.677	2.482	2.011
	Reno-BBR	0.000	3.586	1.658	1.537

4.5. Mixed Scenario

Finally, a mixed scenario is considered to provide an additional validation point for the trends observed in the previous analyses. This case combines the asymmetric dual-backhaul setting with heterogeneous traffic sources, so as to assess whether the behaviors identified under more structured conditions remain visible when different transport demands coexist over the same backhaul environment. More specifically, the mixed scenario includes, for each protocol stack, 3 high-capacity UEs and 3000 low-capacity UEs, thereby generating a composite load profile composed of a limited number of high-rate users and a much larger population of low-rate devices. This final experiment is carried out considering the BBR-BBR and Reno-BBR pairings, since these configurations emerged in the previous analysis as the most representative jitter-efficient solutions in the asymmetric setting. Across the two selected pairings, the mixed scenario confirms the same qualitative pattern already observed in the asymmetric analyses: MPTCP retains the higher throughput contribution on both backhaul links, whereas MPQUIC generally operates at lower average protocol-level jitter, with TN2 remaining the more penalized path.

Under the BBR-BBR pairing (Figure 19), MPTCP retains a clear throughput advantage over MPQUIC on both terrestrial backhaul links also in the mixed scenario. As shown in Figure 19a, on TN1, MPTCP reaches $\mu = 318.24$ Mbps, whereas MPQUIC operates at $\mu = 240.00$ Mbps. On TN2, both protocol stacks experience a throughput reduction, but the same ordering is preserved, with MPTCP at $\mu = 176.28$ Mbps and MPQUIC at $\mu = 146.81$ Mbps. Therefore, even under more composite operating conditions, the BBR-BBR pairing still enables the TCP-based stack to exploit both backhaul links more effectively, while MPQUIC remains at lower throughput regions. Figure 19b reports the protocol-independent jitter comparison between the two backhaul links. The two distributions remain concentrated at low values, but TN2 is visibly shifted toward higher jitter and exhibits a broader spread than TN1, with $\mu = 1.821$ ms versus $\mu = 1.524$ ms. Figure 19c,d isolate protocol-specific jitter on TN1 and TN2, respectively. On TN1, MPQUIC remains centered at significantly lower jitter values than MPTCP, with $\mu = 2.020$ ms compared with $\mu = 4.211$ ms for the TCP-based stack. A similar separation is observed on TN2, where MPQUIC reaches $\mu = 2.493$ ms while MPTCP is shifted to a higher operating region with $\mu = 5.012$ ms. Hence, although MPTCP preserves a clear throughput advantage on both links, MPQUIC continues to provide lower average protocol-level jitter, while the more penalized TN2 path increases the jitter of both stacks with respect to TN1.

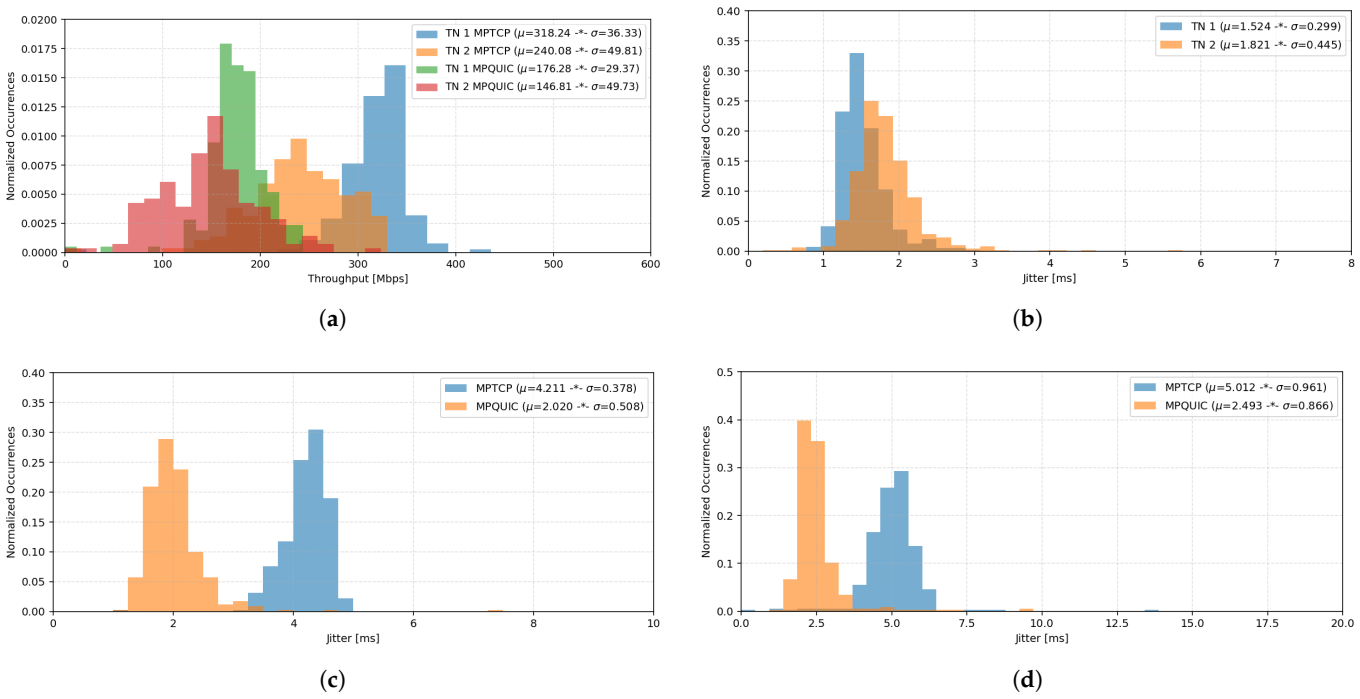

Figure 19. MPTCP and MPQUIC throughput and jitter distributions in mixed scenario under the BBR-BBR CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

In Figure 20a, the Reno-BBR pairing preserves the same qualitative behavior observed for BBR-BBR, namely a consistent throughput advantage of MPTCP over MPQUIC on both terrestrial backhaul links. On TN1, MPTCP reaches $\mu = 313.81$ Mbps, whereas MPQUIC operates at $\mu = 256.02$ Mbps. On TN2, the separation remains evident, with MPTCP at $\mu = 178.62$ Mbps and MPQUIC dropping to $\mu = 129.75$ Mbps. Therefore, also in the mixed scenario, the TCP-based stack continues to exploit both backhaul links more effectively, while the QUIC-based stack remains more penalized on the second link. Figure 20b reports the protocol-independent jitter comparison between the two backhaul links. As in the BBR-BBR case, the two distributions remain concentrated at low values, but TN2 is again slightly shifted toward higher jitter and exhibits a somewhat broader spread than TN1, with $\mu = 1.858$ ms versus $\mu = 1.682$ ms. Figure 20c,d isolate protocol-specific jitter on TN1 and TN2, respectively. On TN1, MPQUIC remains centered at lower average jitter values than MPTCP, with $\mu = 2.258$ ms compared with $\mu = 4.501$ ms for the TCP-based stack. A similar separation is observed on TN2, where MPQUIC reaches $\mu = 2.576$ ms while MPTCP is shifted to a higher operating region with $\mu = 5.326$ ms. Hence, even under Reno-BBR, MPQUIC preserves lower average protocol-level jitter, whereas MPTCP remains associated with higher jitter values but also with the dominant throughput contribution on both backhaul links. Compared with BBR-BBR, however, Reno-BBR produces slightly higher jitter values at both link and protocol level, without providing a clear throughput benefit.

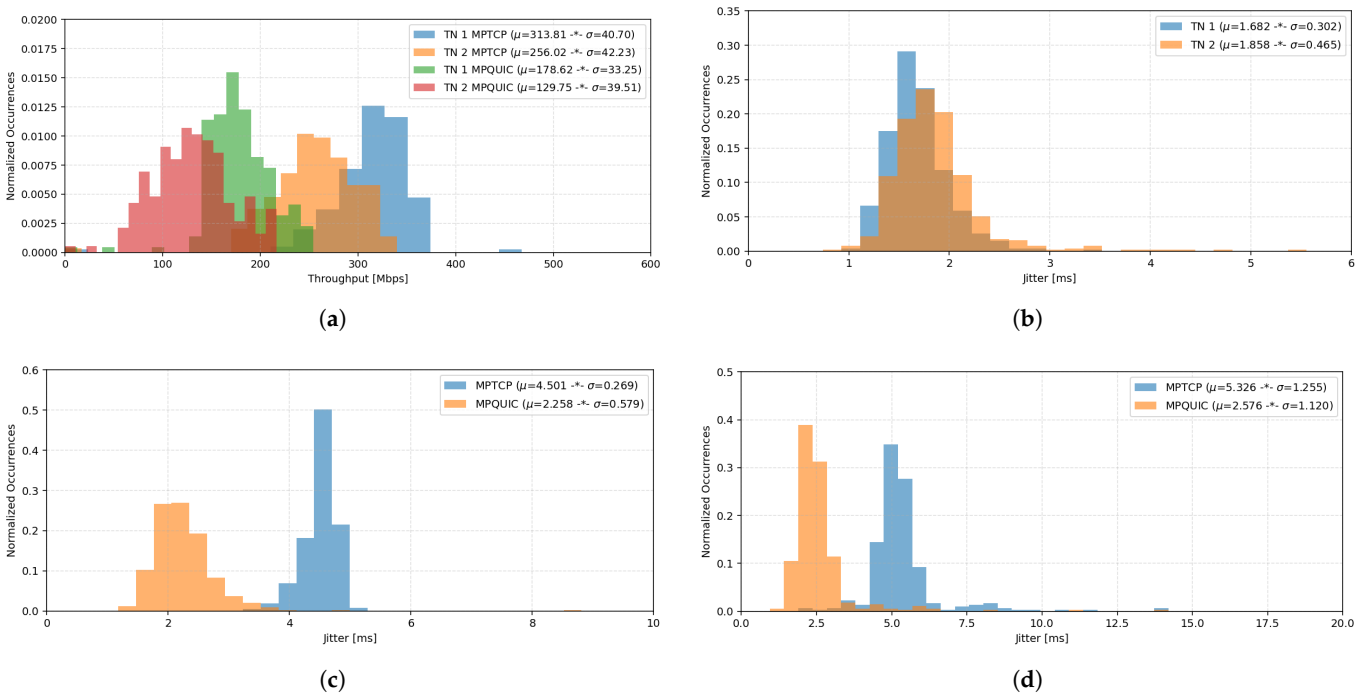

Figure 20. MPTCP and MPQUIC throughput and jitter distributions in mixed scenario under the Reno-BBR CC. Specifically: (**a**) throughput distributions on TN1 and TN2; (**b**) aggregate jitter distributions on TN1 and TN2; (**c**) protocol-specific jitter distributions on TN1; (**d**) protocol-specific jitter distributions on TN2.

Table 20 summarizes the mean jitter values observed for the two CC pairings considered in the mixed scenario.

Table 20. Summary of the mean jitter values extracted from Figures 19 and 20 for the mixed scenario.

MPTCP/MPQUIC CC	TN1	TN2	TN1 MPTCP	TN1 MPQUIC	TN2 MPTCP	TN2 MPQUIC
BBR-BBR	**1.524**	**1.821**	**4.211**	**2.020**	**5.012**	**2.493**
Reno-BBR	1.682	1.858	4.501	2.258	5.326	2.576

From the comparison between Figures 19 and 20, and Table 20, BBR-BBR emerges as the most favorable configuration for the mixed scenario. The two combinations provide a very similar throughput ordering between MPTCP and MPQUIC, with only marginal differences in the mean throughput values on TN1 and TN2. However, BBR-BBR achieves systematically lower link-level jitter and lower protocol-specific jitter on both backhaul links, especially on TN2, which remains the more critical path in the asymmetric setting. These simulations therefore further confirm the robustness of the BBR-BBR pairing as the most suitable reference configuration for the concluding mixed scenario, although the improvement over Reno-BBR remains marginal. In particular, BBR-BBR combines stable throughput separation with the lowest overall delay variability among the two considered options.

Tables 21 and 22 show a very similar throughput ordering for the two tested pairings, with BBR-BBR remaining slightly ahead overall. In this final scenario, unlike the previous eMBB cases, no alternative configuration overturns that ranking from the overall-comparison standpoint, so BBR-BBR remains the most convincing reference solution. Therefore, here the pairing with the best throughput behavior also remains the preferred overall option.

Table 21. Summary of MPTCP throughput statistics for the mixed scenario.

Path	CC	Min	Max	μ	σ
TN1	BBR-BBR	216.84	432.57	318.24	36.33
	Reno-BBR	223.91	446.18	313.81	40.70
TN2	BBR-BBR	134.26	327.94	240.08	49.81
	Reno-BBR	148.73	334.61	256.02	42.23

Table 22. Summary of MPQUIC throughput statistics for the mixed scenario.

Path	CC	Min	Max	μ	σ
TN1	BBR-BBR	38.42	257.63	176.28	29.37
	Reno-BBR	41.17	262.84	178.62	33.25
TN2	BBR-BBR	11.36	296.42	146.81	49.73
	Reno-BBR	14.28	271.53	129.75	39.51

Table 23 confirms and further strengthens the conclusion already drawn for the mixed scenario. Among the two considered pairings, BBR-BBR achieves lower packet loss than Reno-BBR for both MPTCP and MPQUIC, with a particularly visible reduction in the MPTCP case. This result is fully consistent with the mixed-scenario discussion, where BBR-BBR had already emerged as the most favorable overall configuration because it combines very similar throughput ordering with systematically lower jitter. The packet-loss statistics now show that this advantage is not limited to delay stability, but also extends to reliability. Therefore, in the mixed scenario, BBR-BBR should not only be regarded as a good compromise, but rather as the most convincing overall choice among the two tested pairings, since it combines competitive throughput, the lowest jitter, and the lowest packet-loss profile. By contrast, Reno-BBR does not provide a sufficiently clear throughput gain to compensate for its higher jitter and higher loss levels.

Table 23. Summary of packet loss statistics for the mixed scenario.

Path	CC	Min	Max	μ	σ
MPTCP	BBR-BBR	0.000	0.072	0.028	0.020
	Reno-BBR	0.000	0.594	0.214	0.197
MPQUIC	BBR-BBR	0.000	0.351	0.188	0.128
	Reno-BBR	0.000	0.603	0.296	0.225

5. Conclusions

This paper investigated the impact of CC choices on multipath transport over a dual-link terrestrial backhaul, considering a 5G IAB scenario in which MPTCP-enabled and MPQUIC-enabled UEs concurrently share two terrestrial backhaul paths toward the core network. Within a Mininet-based experimental framework, all feasible cross-combinations of BBR, CUBIC, and Reno across the MPTCP and MPQUIC stacks were evaluated in order to assess how CC dynamics affect throughput sharing, fairness, jitter, and packet loss under shared backhaul conditions.

The obtained results show that the selected CC pairing has a decisive impact on cross-protocol coexistence. In most configurations, MPTCP achieves higher throughput than MPQUIC on both backhaul links, confirming a general advantage of the TCP-based stack in terms of effective path exploitation. However, the extent of this advantage strongly depends on the adopted pairing. From a fairness perspective, the symmetric dual-link scenario highlights BBR-CUBIC as the best option in terms of aggregate throughput balance between

MPTCP and MPQUIC, whereas CUBIC-CUBIC provides the most balanced per-link fairness across the two terrestrial backhaul paths. In the asymmetric scenario, BBR-CUBIC emerges as the most favorable configuration also from the fairness standpoint, since it achieves the highest Jain's fairness index on both links and on the aggregate throughput.

The jitter analysis further shows that throughput balance and delay stability do not necessarily coincide. In the symmetric scenario, the most favorable jitter-oriented solutions are Reno-BBR and CUBIC-BBR, whereas in the asymmetric scenario the best jitter-oriented pairings are Reno-BBR and BBR-BBR. This confirms that a configuration that improves fairness or throughput balance may still induce substantially larger protocol-level jitter, especially for MPQUIC, whose distributions are often broader and more heavy-tailed than those of MPTCP. The analyses under mMTC-inspired and mixed traffic conditions confirm the same fundamental trade-off under more demanding operating regimes. As the number of active low-rate users increases, the cross-protocol imbalance becomes more evident, and the two transport stacks tend to specialize over different backhaul links rather than jointly exploiting both paths in a balanced way. In the asymmetric mMTC-inspired case, BBR-BBR remains slightly more favorable from a throughput perspective, whereas Reno-BBR provides the best overall jitter behavior. In the final mixed scenario, BBR-BBR confirms itself as the most robust configuration among the considered options, since it preserves stable throughput ordering between MPTCP and MPQUIC while also maintaining the lowest overall delay variability. The packet-loss analysis reinforces the same overall interpretation: the pairings that are most favorable in throughput or fairness terms are not always the ones minimizing loss, whereas configurations such as CUBIC-BBR, Reno-BBR, and, in the mixed scenario, BBR-BBR emerge as the most balanced compromises when throughput, delay stability, and reliability are considered jointly.

Overall, the study demonstrates that CC selection is a first-order design dimension for MPTCP/MPQUIC-based multi-connectivity over terrestrial multi-link backhauls. No single pairing simultaneously optimizes throughput sharing, fairness, and jitter stability. Instead, the results show that transport-level control choices act as practical distributed resource-allocation mechanisms, directly influencing efficiency, fairness, stability, and scalability in heterogeneous multi-link infrastructures. Consequently, protocol evaluation in these environments must explicitly account for the trade-offs among throughput, fairness, and delay variability. Future work will extend the analysis to additional CC variants and tunings, broader traffic mixes, and larger-scale or hardware-based experimental platforms, in order to further characterize the design space of multipath transport over heterogeneous terrestrial backhaul systems and to support more informed transport-layer design choices in multi-link 5G and beyond networks.

Author Contributions: Conceptualization, R.P. and D.T.; methodology, R.P.; software, R.P.; validation, R.P. and D.T.; formal analysis, R.P. and D.T.; investigation, R.P.; resources, D.T.; data curation, R.P.; writing—original draft preparation, R.P.; writing—review and editing, R.P. and D.T.; visualization, R.P.; supervision, D.T.; project administration, D.T.; funding acquisition, D.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data is contained within the article.

Acknowledgments: Generative AI tools were used during manuscript preparation for language editing, text rephrasing, and structural refinement. All scientific content, experimental design, analysis, interpretation of results, and final manuscript verification were performed and approved by the authors, who take full responsibility for the content of the submitted work.

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

References

1. Multipath TCP Project. Multipath TCP (MPTCP) Community Website. Available online: https://www.mptcp.dev/ (accessed on 1 February 2026).
2. Ford, A.; Raiciu, C.; Handley, M.J.; Bonaventure, O.; Paasch, C. TCP Extensions for Multipath Operation with Multiple Addresses. RFC 8684 2020. Available online: https://www.rfc-editor.org/rfc/rfc6824.html (accessed on 1 February 2026).
3. Pokhrel, S.R.; Mandjes, M. Improving Multipath TCP Performance over WiFi and Cellular Networks: An Analytical Approach. *IEEE Trans. Mob. Comput.* **2019**, *18*, 2562–2576. [CrossRef]
4. Fan, W.; Pan, Y.; Xiao, F.; Lv, M.; Han, L.; Yu, S. Reliable and Efficient Multi-Path Transmission Based on Disjoint Paths in Data Center Networks. *IEEE Trans. Comput.* **2025**, *74*, 3362–3376. [CrossRef]
5. Multipath QUIC Project. Multipath QUIC (MP-QUIC) Project Website. Available online: https://multipath-quic.org/ (accessed on 1 February 2026).
6. Iyengar, J.; Thomson, M. QUIC: A UDP-Based Multiplexed and Secure Transport. RFC 9000 2021. Available online: https://www.rfc-editor.org/info/rfc9000 (accessed on 1 February 2026).
7. De Coninck, Q.; Bonaventure, O. Multipath QUIC: Design and Evaluation. In *Proceedings of the 13th International Conference on Emerging Networking EXperiments and Technologies*; CoNEXT '17; Association for Computing Machinery: New York, NY, USA, 2017; pp. 160–166. [CrossRef]
8. 3GPP. NR; Study on Integrated Access and Backhaul. Technical Report TR 38.874, 3rd Generation Partnership Project (3GPP), 2019. Release 16. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3232 (accessed on 1 February 2026).
9. ETSI. 5G; NG-RAN; Architecture Description (3GPP TS 38.401). Technical Specification ETSI TS 138 401, European Telecommunications Standards Institute (ETSI), 2025. Version 18.7.0 (2025-10). Available online: https://www.etsi.org/deliver/etsi_ts/138400_138499/138401/18.07.00_60/ts_138401v180700p.pdf (accessed on 1 February 2026).
10. ETSI. 5G; NR; Backhaul Adaptation Protocol (BAP) specification (3GPP TS 38.340). Technical Specification ETSI TS 138 340, European Telecommunications Standards Institute (ETSI), 2025. Version 19.0.0 (2025-10). Available online: https://www.etsi.org/deliver/etsi_ts/138300_138399/138340/19.00.00_60/ts_138340v190000p.pdf (accessed on 1 February 2026).
11. Polese, M.; Giordani, M.; Zugno, T.; Roy, A.; Goyal, S.; Castor, D.; Zorzi, M. Integrated Access and Backhaul in 5G mmWave Networks: Potential and Challenges. *IEEE Commun. Mag.* **2020**, *58*, 62–68. [CrossRef]
12. Cudak, M.; Ghosh, A.; Ghosh, A.; Andrews, J. Integrated Access and Backhaul: A Key Enabler for 5G Millimeter-Wave Deployments. *IEEE Commun. Mag.* **2021**, *59*, 88–94. [CrossRef]
13. Madapatha, C.; Makki, B.; Fang, C.; Teyeb, O.; Dahlman, E.; Alouini, M.S.; Svensson, T. On Integrated Access and Backhaul Networks: Current Status and Potentials. *IEEE Open J. Commun. Soc.* **2020**, *1*, 1374–1389. [CrossRef]
14. Nguyen, C.T.; Saputra, Y.M.; Huynh, N.V.; Nguyen, T.N.; Hoang, D.T.; Nguyen, D.N.; Pham, V.Q.; Voznak, M.; Chatzinotas, S.; Tran, D.H. Emerging Technologies for 6G Non-Terrestrial-Networks: From Academia to Industrial Applications. *IEEE Open J. Commun. Soc.* **2024**, *5*, 3852–3885. [CrossRef]
15. Chiti, F.; Picchi, R.; Pierucci, L. From Earth-to-Moon Networking: A Software-Defined Temporal Perspective. *IEEE Trans. Netw. Sci. Eng.* **2025**, *12*, 369–380. [CrossRef]
16. 3GPP. 5G; 5G System; Access Traffic Steering, Switching and Splitting (ATSSS); Stage 3. Technical Specification TS 24.193, 3rd Generation Partnership Project (3GPP), 2021. Release 16. Available online: https://www.etsi.org/deliver/etsi_ts/124100_124199/124193/16.02.00_60/ts_124193v160200p.pdf (accessed on 1 February 2026).
17. Cogalan, T.; Kheirkhah, M.; Patani, K.; Camps-Mur, D.; Mourad, A. Enhanced Access Traffic Steering Splitting Switching with Utility-Based Decisioning. In *Proceedings of the 2022 IEEE Conference on Standards for Communications and Networking (CSCN)*; IEEE: Piscataway, NJ, USA, 2022; pp. 138–143. [CrossRef]
18. Jiang, H.; Liu, Z.; Wang, Y.; Lee, K.; Rhee, I. Understanding bufferbloat in cellular networks. In *Proceedings of the 2012 ACM SIGCOMM Workshop on Cellular Networks: Operations, Challenges, and Future Design*; CellNet '12; IEEE: New York, NY, USA, 2012; pp. 1–6. [CrossRef]
19. Alfredsson, S.; Del Giudice, G.; Garcia, J.; Brunstrom, A.; De Cicco, L.; Mascolo, S. Impact of TCP congestion control on bufferbloat in cellular networks. In *Proceedings of the 2013 IEEE 14th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*; IEEE: Piscataway, NJ, USA, 2013; pp. 1–7. [CrossRef]
20. Du, P.; Nazari, S.; Mena, J.; Fan, R.; Gerla, M.; Gupta, R. Multipath TCP in SDN-enabled LEO satellite networks. In *Proceedings of the MILCOM 2016–2016 IEEE Military Communications Conference*; IEEE: Piscataway, NJ, USA, 2016; pp. 354–359. [CrossRef]
21. Bockelmann, C.; Pratas, N.; Nikopour, H.; Au, K.; Svensson, T.; Stefanovic, C.; Popovski, P.; Dekorsy, A. Massive machine-type communications in 5g: Physical and MAC-layer solutions. *IEEE Commun. Mag.* **2016**, *54*, 59–65. [CrossRef]
22. Sharma, S.K.; Wang, X. Toward Massive Machine Type Communications in Ultra-Dense Cellular IoT Networks: Current Issues and Machine Learning-Assisted Solutions. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 426–471. [CrossRef]

23. Popovski, P.; Trillingsgaard, K.F.; Simeone, O.; Durisi, G. 5G Wireless Network Slicing for eMBB, URLLC, and mMTC: A Communication-Theoretic View. *IEEE Access* **2018**, *6*, 55765–55779. [CrossRef]
24. Saha, C.; Afshang, M.; Dhillon, H.S. Bandwidth Partitioning and Downlink Analysis in Millimeter Wave Integrated Access and Backhaul for 5G. *IEEE Trans. Wirel. Commun.* **2018**, *17*, 8195–8210. [CrossRef]
25. Pagin, M.; Zugno, T.; Polese, M.; Zorzi, M. Resource Management for 5G NR Integrated Access and Backhaul: A Semi-Centralized Approach. *IEEE Trans. Wirel. Commun.* **2022**, *21*, 753–767. [CrossRef]
26. Gopalam, S.; Hanly, S.V.; Whiting, P. Distributed Resource Allocation and Flow Control Algorithms for mmWave IAB Networks. *IEEE/ACM Trans. Netw.* **2023**, *31*, 3175–3190. [CrossRef]
27. Pueyo, J.; Camps-Mur, D.; Catalan-Cid, M. PHaul: A PPO-Based Forwarding Agent for Sub6-Enhanced Integrated Access and Backhaul Networks. *IEEE Trans. Netw. Serv. Manag.* **2024**, *21*, 6273–6289. [CrossRef]
28. Zhang, B.; Filippini, I. Mobility-Aware Resource Allocation for mmWave IAB Networks: A Multi-Agent Reinforcement Learning Approach. *IEEE/ACM Trans. Netw.* **2024**, *32*, 3559–3574. [CrossRef]
29. Ding, L.; Tian, Y.; Liu, T.; Wei, Z.; Zhang, X. Understanding commercial 5G and its implications to (Multipath) TCP. *Comput. Netw.* **2021**, *198*, 108401. [CrossRef]
30. Poorzare, R.; Waldhorst, O.P. Toward the Implementation of MPTCP Over mmWave 5G and Beyond: Analysis, Challenges, and Solutions. *IEEE Access* **2023**, *11*, 19534–19566. [CrossRef]
31. Xing, Y.; Xue, K.; Zhang, Y.; Han, J.; Li, J.; WeiMember, D.S.L. An Online Learning Assisted Packet Scheduler for MPTCP in Mobile Networks. *IEEE/ACM Trans. Netw.* **2023**, *31*, 2297–2312. [CrossRef]
32. Yang, W.; Cai, L.; Shu, S.; Pan, J.; Sepahi, A. MAMS: Mobility-Aware Multipath Scheduler for MPQUIC. *IEEE/ACM Trans. Netw.* **2024**, *32*, 3237–3252. [CrossRef]
33. Das, S.; Guo, J.; Parsons, K.; Nagai, Y.; Sumi, T.; Sakaguchi, N.; Orlik, P.; Kalafatis, S. Learning Based Scheduling and Adaptive Congestion Control for Multipath QUIC. In *Proceedings of the 2025 IEEE International Conference on Communications Workshops (ICC Workshops)*; IEEE: Piscataway, NJ, USA, 2025; pp. 830–836. [CrossRef]
34. Miuccio, L.; Riolo, S.; Samarakoon, S.; Bennis, M.; Panno, D. Emerging Generalized Wireless MAC Communication Protocols via Abstraction. *IEEE Open J. Commun. Soc.* **2025**, *6*, 6842–6865. [CrossRef]
35. Trúchly, P.; Laš, M.; Bencel, R.; Kuntová, H. Software-Defined LEO Satellite Networks with Focus on Multipath TCP Data Throughput. In *Proceedings of the 2022 International Symposium ELMAR*; IEEE: Piscataway, NJ, USA, 2022; pp. 149–152. [CrossRef]
36. Chiti, F.; Picchi, R.; Pierucci, L. Advanced Control Architectures for Quantum Satellite Temporal-Networking. *IEEE Access* **2024**, *12*, 43410–43421. [CrossRef]
37. Chiti, F.; Pecorella, T.; Picchi, R.; Pierucci, L. Towards Digital-Twin Assisted Software-Defined Quantum Satellite Networks. *Sensors* **2025**, *25*, 889. [CrossRef] [PubMed]
38. Li, H.; Sun, C.; Cui, T.; Wang, S. 5G Integrated Access and Backhaul: Performance Analysis of Congestion Control in 3GPP. In *Proceedings of the 2024 IEEE 99th Vehicular Technology Conference (VTC2024-Spring)*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–6. [CrossRef]
39. Sadovaya, Y.; Vikhrova, O.; Mao, W.; Semiari, O.; Yeh, S.P.; Nikopour, H.; Talwar, S.; Andreev, S. Distributed Delay-Aware Link Scheduling and Route Selection in mmWave IAB Networks. In *Proceedings of the GLOBECOM 2024–2024 IEEE Global Communications Conference*; IEEE: Piscataway, NJ, USA, 2024; pp. 391–396. [CrossRef]
40. Zhivtsova, A.; Beschastnyi, V.; Koucheryavy, Y.; Samouylov, K. A Survey of Delay-Oriented Dynamic Link Scheduling Policies for 5G/6G Integrated Access and Backhaul Systems. *IEEE Access* **2024**, *12*, 118565–118586. [CrossRef]
41. Austria, P.; Park, C.H.; Jo, J.Y.; Kim, Y.; Sundaresan, R.; Pham, K. BBR Congestion Control Analysis with Multipath TCP (MPTCP) and Asymmetrical Latency Subflow. In *Proceedings of the 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*; IEEE: Piscataway, NJ, USA, 2022; pp. 1065–1069. [CrossRef]
42. Kanaya, T.; Tabata, N.; Yamaguchi, S. A Study on Performance of CUBIC TCP and TCP BBR in 5G Environment. In *Proceedings of the 2020 IEEE 3rd 5G World Forum (5GWF)*; IEEE: Piscataway, NJ, USA, 2020; pp. 508–513. [CrossRef]
43. Ware, R.; Mukerjee, M.K.; Seshan, S.; Sherry, J. Modeling BBR's Interactions with Loss-Based Congestion Control. In *Proceedings of the Proceedings of the Internet Measurement Conference*; IMC '19; IEEE: New York, NY, USA, 2019; pp. 137–143. [CrossRef]
44. Nguyen, T.K.; Nguyen, C.T.; Le, H.D.; Pham, A.T. TCP Performance Over Satellite-Based Hybrid FSO/RF Vehicular Networks: Modeling and Analysis. *IEEE Access* **2021**, *9*, 108426–108440. [CrossRef]
45. Le, H.D.; Trinh, P.V.; Pham, T.V.; Kolev, D.R.; Carrasco-Casado, A.; Kubo-Oka, T.; Toyoshima, M.; Pham, A.T. Throughput Analysis for TCP Over the FSO-Based Satellite-Assisted Internet of Vehicles. *IEEE Trans. Veh. Technol.* **2022**, *71*, 1875–1890. [CrossRef]
46. Mininet Project. Mininet: Instant Virtual Networks on Your Laptop (or Other PC). Available online: https://mininet.org/ (accessed on 1 February 2026).
47. The Linux Kernel Documentation. Linux Networking Filter (BPF/eBPF). Online Documentation. Available online: https://docs.kernel.org/networking/filter.html (accessed on 1 February 2026).

48. eBPF Foundation. What is eBPF? Available online: https://ebpf.io/what-is-ebpf/ (accessed on 6 February 2026).
49. iPerf Project. iPerf: Network Bandwidth Measurement Tool. Available online: https://iperf.fr/ (accessed on 1 February 2026).
50. Private Octopus Inc. picoquic: A Minimal Implementation of the QUIC Protocol. GitHub Repository. Available online: https://github.com/private-octopus/picoquic (accessed on 27 February 2026).
51. Blanton, E.; Paxson, D.V.; Allman, M. TCP Congestion Control. RFC 5681 2009. Available online: https://www.rfc-editor.org/info/rfc5681 (accessed on 1 February 2026).
52. Xu, L.; Ha, S.; Rhee, I.; Goel, V.; Eggert, L. CUBIC for Fast and Long-Distance Networks. RFC 9438 2023. Available online: https://www.rfc-editor.org/info/rfc9438 (accessed on 1 February 2026).
53. Cardwell, N.; Cheng, Y.; Yeganeh, S.H.; Swett, I.; Jacobson, V. BBR Congestion Control. *Internet-Draft Draft-Cardwell-Iccrg-Bbr-Congestion-Control-02*; Work in Progress; Internet Engineering Task Force: Wilmington, DE, USA, 2022.
54. Multipath TCP. mptcp_net-next: Development version of the Upstream MultiPath TCP Linux kernel. GitHub Repository. Available online: https://github.com/multipath-tcp/mptcp_net-next (accessed on 1 February 2026).
55. Alnanih, R.; Elrefaei, L.; Al-Ahwal, A. Advancing Sustainability Through an IoT-Driven Smart Waste Management System with Software Engineering Integration. *Sustainability* **2025**, *17*, 9803. [CrossRef]
56. Lubna, T.; Mahmud, I.; Cho, Y.Z. Low Latency and High Data Rate (LLHD) Scheduler: A Multipath TCP Scheduler for Dynamic and Heterogeneous Networks. *Sensors* **2022**, *22*, 9869. [CrossRef]
57. Lo Bello, L.; Steiner, W. A Perspective on IEEE Time-Sensitive Networking for Industrial Communication and Automation Systems. *Proc. IEEE* **2019**, *107*, 1094–1120. [CrossRef]