To Poly or Not to Poly: QoS-Aware Multi-Path Networking

Ricardo Andres Quiceno*, Tobias Fischer†, Zubair Shaik*†, and Andreas Mitschele-Thiel*†
*Dept. of Computer Science, TU Ilmenau, Germany; †AiVader GmbH, Germany
{ricardo-andres.quiceno-betancur, tobias.fischer, mitsch}@tu-ilmenau.de,
zubair.shaik@aivader.com

***Abstract*—In today's world of growing network demands, high data rate, reliability, and low delay are crucial for seamless user experiences. Multipath networking offers a possible solution by optimizing existing infrastructure, aggregating multiple – possibly different (heterogeneous) – connections (e.g., Wi-Fi, 3GPP- Cellular, Ethernet) into a single, efficient data stream. This is realized by distributing traffic across multiple paths, aiming to reduce the impact of bottlenecks. However – to the best of our knowledge – current solutions often require changing application sources or disrupt end-to-end control. This paper introduces a Multipath Function (MPF)-Gateway, built on our Contextual Hybrid Protocol for Multipath (CHYMP), which enhances network performance while keeping existing control mechanisms intact. By integrating an innovative Quality-of-Service (QoS) Manager, our gateway meets application-specific demands, making it ideal for diverse traffic in heterogeneous networks. With reconfigurable schedulers and low-latency options, CHYMP adapts to dynamic policies, offering a scalable and flexible solution for hybrid networks. The combination of our MPF- Gateway and CHYMP advances multipath systems, improving efficiency, stability, and adaptability with increased control in our comparison to alternative solutions.**

***Index Terms*—Heterogeneous networks, Mobile communication, Quality of service, Protocols, TCPIP**

I. Introduction

As modern safety systems increasingly depend on network-controlled architectures, single-path communication protocols face limitations. While advancements in 3GPP 5G and IEEE 802.11 WLAN extend connectivity, isolated operation restricts their potential in heterogeneous networks.

A prominent example is railway logistics modernization. The Future Railway Mobile Communication System (FRMCS) [1], developed under the 3GPP initiative, integrates 3GPP-5G, replacing GSM-R to enable resilient, high-speed communication necessary for safe, remote operations.

A. Limitations of Single-Path Protocols

Conventional transport protocols, optimized for single-path transfer, constrain multipath networks by utilizing only one link, limiting throughput and resilience. Multipath protocols aim to overcome this by distributing data across multiple paths, enhancing resource use and fault tolerance. However, many multipath protocols require extensive client-side modifications, posing deployment challenges similar to the IPv4-to-IPv6 transition [2].

Multipath translation gateways offer an alternative, translating single-path protocols into multipath ones. However, these gateways also introduce unique challenges.

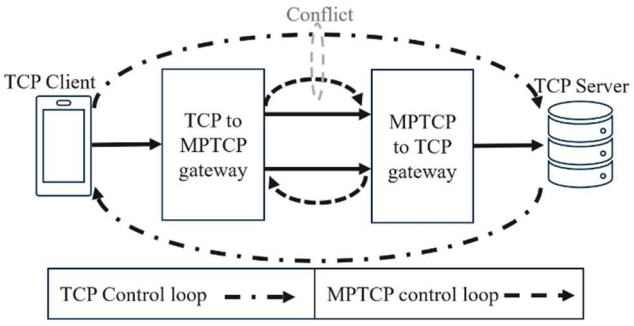

Figure 1. Control loop issues in basic translation gateways.

B. Challenges with Multipath Translation Gateways

Multipath translation gateways, as shown in Figure 1, often introduce conflicting control loops, destabilizing end- to-end performance, particularly when protocols like TCP and Multipath-TCP (MPTCP) coexist. Both protocols independently manage retransmission, congestion control, and acknowledgments, leading to feedback loops that can oscillate and interfere with each other. These overlapping control mechanisms reduce network efficiency, as a single active control function is sufficient to manage the entire path effectively, avoiding redundant operations.

C. Need for Quality-of-Service (QoS) Management

Current multipath protocols and gateways generally lack QoS management, which limits their effectiveness for critical applications requiring low latency and high reliability – which is essential for effective use of heterogeneous networks.

D. Proposed Solutions

This paper introduces two complementary solutions:

1) **Contextual Hybrid Protocol for Multipath (CHYMP)**: A novel, UDP-based multipath protocol that addresses control loop conflicts in traditional gateways. *CHYMP* includes QoS features through a custom header, enabling efficient data flow across paths.
2) **Multipath Function (MPF)-Gateway**: Our gateway solution works with *CHYMP* to provide comprehensive multipath and QoS management, dynamically prioritizing traffic to optimize resource usage based on application-specific needs.

ISBN: 978-3-903176-71-3

II. Inspiration for the solutions

A. Modern Single-Path Solutions

The limitations of TCP in meeting the demands of modern, high-performance web applications prompted the development of **Quick UDP Internet Connections (QUIC)**, a "multiplexed and secure transport protocol that runs on top of UDP" [3]. Originally developed by Google and later standardized by the Internet Engineering Task Force (IETF), QUIC is optimized for HTTP/3 traffic and offers key improvements, including:

- Reduced connection setup time
- Enhanced congestion control mechanisms
- Secure communication through QUIC-TLS encryption
- Support for network path migration

These features address issues like head-of-line blocking, significantly enhancing user experience by enabling low-latency, resilient connections across varying network conditions.

A notable advancement in QUIC is the **datagram mode**, proposed in [4]. This extension enables a simplified, unreliable transport mode without acknowledgments or retransmissions, making QUIC suitable for real-time applications such as voice and video, where low latency is crucial.

Another remarkable advancement in single-path networking is the **Stream Control Transport Protocol (SCTP)** [5]. This protocol promises reliable transmission, while also providing redundancy through backup links, which will only be used in case of failures, commonly described as "Failover". To detect network problems, SCTP uses "Heartbeats" – control messagesthat are sent periodically – to confirm route availability. SCTPgained widespread attention, leading to implementation in the 3GPP 4G and 5G standards [6].

B. Protocol Evolutions for Multipath

Multipath protocols facilitate data transmission across multiple network paths simultaneously, providing load balancing, resource pooling, and resilience. Key components of these protocols include connection setup, path management, sequence numbering, authentication, and congestion control, managed by three primary functional blocks: the path manager, scheduler,and congestion controller.

Both **MPTCP** and **Multipath-QUIC (MPQUIC)** were foundational in guiding the design of *CHYMP* and our *MPF-Gateway*.

1) MPTCP: An extension of TCP enabling multiple paths to be used concurrently, enhancing both bandwidth utilization and fault tolerance while addressing head-of-line blocking. Theprimary characteristics of MPTCP are:

- **Connection Establishment:**
  MPTCP initiates with a primary connection over TCP, while additional subflows (paths) can be added dynamically, maintaining backward compatibility with TCP.
- **Path Addition:**
  Dynamic path addition allows multiple subflows to operate within the same logical connection, increasing throughput and/or reliability.
- **Sequence Numbers and Number Spaces:**
  Each subflow has its own sequence numbering, with the primary connection overseeing overall sequencing, ensuring accurate packet reassembly.
- **Authentication and Security:**
  MPTCP employs hash-based message authentication code (HMAC)-based authentication to secure sessions and prevent unauthorized path inclusion, alongside sessionhijacking protections.

2) MPQUIC: An evolution of the QUIC protocol, that introduces new message types specifically designed for connection establishment, the configuration of additional paths, and overall path management [7]. One important feature in MPQUIC is theNumber spaces, that provide reliable in-order delivery through different numbering for data and path.

The formal MPQUIC specification [7] and current implemen- tations lack a framework to incorporate the datagram extensionfor multipath contexts – a motivation for us to develop *CHYMP*.

C. Multipath Proxies

Multipath proxies enable multipath capabilities without requiring modifications to the user or server side, easing the adoption of multipath protocols in existing infrastructures. Several configurations, illustrated in Figure 2, address these complexities. Multiple implementations are already available, e.g. concerning TCP and MPTCP: **Transparent Multipath** [8] and the **MPTCP Proxy** [9].

A possible approach is employing the *SOCKS5* protocol, in which the client sends the server's address to the proxy. The proxy then authenticates and connects to the server, mapping traffic between TCP and MPTCP, adding or stripping headers (e.g., MP_CAPABLE) as required by traffic direction.

While proxy-based solutions manage connection setup, data transfer, and termination, they depend on implementation-specific methods, such as proprietary approaches to congestion control.

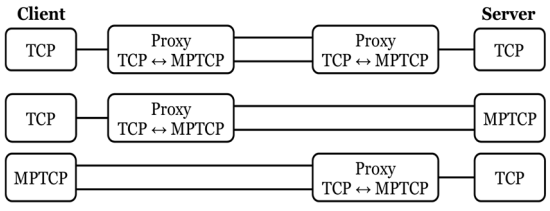

Figure 2. Possible proxy configuration for TCP and MPTCP

D. Access Traffic Steering, Switching, and Splitting (ATSSS)

ATSSS is a 3GPP 5G feature designed to optimize resource use by distributing traffic across multiple networks. Operating on a client-server model, it manages connections at the user equipment (UE) level while "assistance gateways" in the 5G User Plane Function (UPF) support traffic steering based on QoS requirements [6].

ISBN: 978-3-903176-71-3

The ATSSS client dynamically selects paths by monitoring network conditions and QoS parameters, benefiting applications with high demands for low latency, high throughput, and reliability. Its split control-user plane model enables scalable, real-time traffic adjustments.

Within ATSSS, the Performance Management Function (PMF) conducts active and passive traffic measurements, tracking metrics like Round-Trip time (RTT) and packet loss via initiation control messages [10]. This supports traffic adjustments, including Packet Loss Rate (PLR) and access reports, for enhanced management.

Key ATSSS features include:

- Traffic management control
- Multipath protocol selection
- Traffic flow steering, switching, and splitting
- Link performance measurements
- QoS enforcement
- Operational modes:
  1) Active-Standby
  2) Smallest Delay
  3) Load-Balancing
  4) Priority-Based
  5) Redundant

E. Multi-Access Management Systems (MAMS)

MAMS [11] is designed for managing multiple access networks, enabling seamless connectivity and efficient resource utilization. Operating in a client-server model, MAMS is access technology agnostic: the client manages diverse access technologies while the server oversees QoS by employing both active and passive measurements to control traffic flow.

Like ATSSS, MAMS separates control and user planes, allowing for independent path selection and QoS management to improve data handling efficiency. Its core features include:

- Control of traffic management
- Steering, switching, and splitting of traffic flows
- Link performance measurements (active and passive)
- QoS enforcement
- Convergence method selection
- Adaptation method selection

For QoS enforcement, MAMS dynamically monitors and allocates available paths, adding or removing them as necessary to optimize traffic flow and network efficiency.

The convergence method allows defining multipath protocols or encapsulations mechanisms. With adaptation methods, one can further refine user plane parameters for reachability and security configurations, for protocols like Internet Protocol Security (IPSec) and UDP Datagram Transport Layer Security (UDP-DTLS).

MAMS suggests HTTP and JSON for control-plane messaging, offering structure but allowing flexibility in implementation.

In total MAMS presents a solid foundation for protocol development and effective multipath traffic management, even though current implementations are limited.

III. Design of the MPF-Gateway

The *MPF-Gateway* system was developed to provide multipath-enabled service with integrated QoS for applications that require both capabilities. As shown in Figure 3, the system consists of two MPF instances – one on the client-side and one on the server-side – connected through a private and a public 3GPP 5G network in this example.

Both instances are supported by Network-Address-Translation (NAT) and buffer components. These components function as the entry and exit traffic points, enabling traffic classification and temporary storage.

The following shows more details for each component:

- **NAT:**
  This component translates user traffic, tags it appropriately, and processes it for MPF utilization, performing an IP and port mapping into a port-based mapping for seamless integration.
- **Buffer:**
  The buffer temporarily stores tagged traffic before it is consumed by the MPF Client, ensuring smooth transmission flow.
- **MPF:**
  As the core of our gateway, the MPF adds intelligence and multipath capability, with distinct roles at both client and server ends.

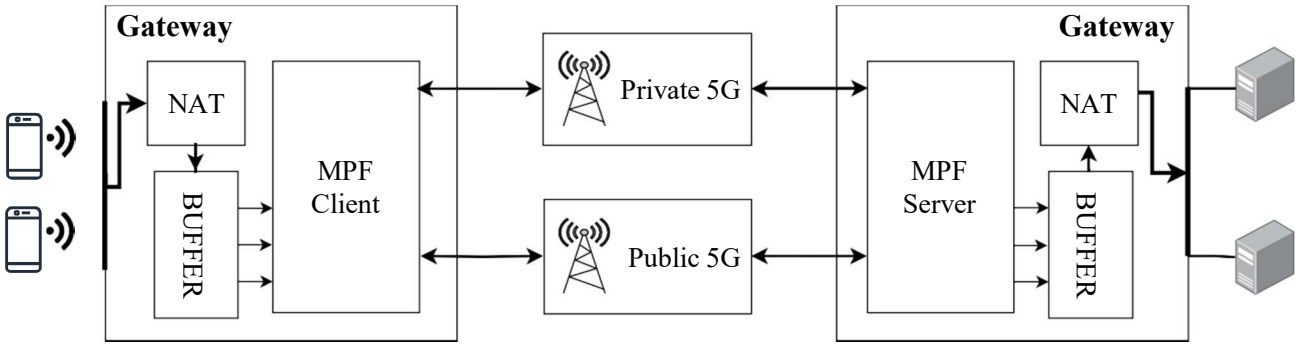

Figure 3. Simplified MPF system with Public and Private 5G as examples

ISBN: 978-3-903176-71-3

The MPF architecture is designed with a control plane requiring a control protocol to manage communication between client and server, facilitating session parameter negotiation.The Gateway can manage multiple concurrent sessions, with each client receiving a specific policy, multipath protocol, and QoS configuration upon connection request. All parameters are negotiated during connection establishment, leading to a tailored user-plane setup.

Key to the system's functionality are its capabilities in steering, switching, and splitting traffic across both the control and user planes. With CHYMP-Support, the MPF enables adaptable user-plane scheduling, supporting options such as:

- **Path selection** for single-path usage,
- **Total failure fallback** with designated primary and backup paths,
- **QoS-compliant fallback** to switch paths during performance issues,
- **Duplication** for high-reliability, low-latency applications by sending packets on multiple paths, and
- **Aggregation** to combine path capacities for increased throughput.

These scheduling modes align with FRMCS requirements, catering to diverse applications needing high reliability (e.g. emergency calls, control commands), high throughput (e.g. monitoring tools), or general-purpose service without strict requirements (e.g., passenger traffic).

Given the system's QoS requirements, specialized modules enable modifications within the user plane. The MPF includes a flexible Multipath Block, allowing selection among various multipath protocols to meet different user demands.

Figure 4 illustrates the final MPF architecture, showcasing these components and capabilities. The roles of each architectural component are defined as follows:

- **Control Manager:**
  Serving as the system's central controller, the Control Manager oversees path management, connection establishment, traffic prioritization, and policy configuration for the user plane, including settings for ports, protocols, security, and interface selection.
- **PMF:**
  Inspired by ATSSS, the PMF handles performance measurement, conducting active tests and collecting passive metrics to support intelligent traffic handling.
- **QoS Manager:**
  The QoS Manager analyzes traffic data collected by the PMF to adjust user traffic policies based on real-time conditions, enabling adaptive quality control.
- **Sending Buffer:**
  This buffer contains the prioritized traffic by type, with a function to adjust send rates to match traffic requirements and system capacity, thus optimizing flow management.
- **MP Protocols:**
  This component manages Layer 4 protocol configurations set by the Control Manager, supporting steering, switching, and splitting for the user plane and enabling diverse scheduling through CHYMP.

Together, these components enable the MPF system to configure gateway operations, manage traffic classification and prioritization, oversee policy management, adapt to QoS demands, ensure reliable and unreliable transmissions, and perform active and passive measurements. The system's advanced capabilities in steering, switching, and splitting traffic across control and data planes position it as a comprehensive multipath solution for complex network environments.

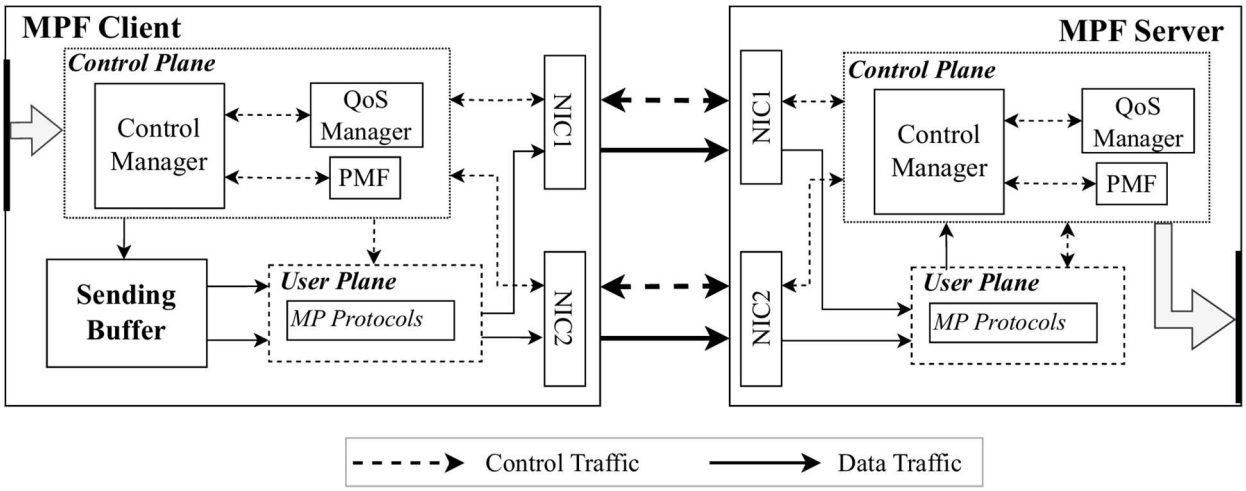

Figure 4. MPF System architecture

ISBN: 978-3-903176-71-3

IV. Design of CHYMP

CHYMP is a multipath protocol designed for heterogeneous networks, providing flexible scheduling and QoS support through real-time path and user-specific metrics. It overcomes traditional multipath limitations with per-path QoS metrics and dynamic adaptability.

CHYMP combines essential multipath functionalities – steering, switching, and splitting – over UDP, supporting controlled (non-standalone) gateway scenarios. The protocol introduces five tailored schedulers optimized for FRMCS traffic requirements, ideal for dynamic network conditions in gateway applications.

The *CHYMP* architecture consists of three main components:

- **Path Manager:**
  Manages connection setup, metrics processing, policy enforcement, path selection, and path monitoring, including path addition/removal.
- **Packet Scheduler:**
  Operates based on policy-based scheduling in non-standalone mode or low-latency scheduling in standalone applications.
- **Packet Handler:**
  Oversees packet reordering, duplication removal, metric measurement, and efficient packet forwarding.

CHYMP provides QoS-Support through its custom header, which includes timestamps and per-path packet numbers. This enables the measurement of throughput, per-path packet loss, and one-way delay, assuming system synchronization. These metrics can be accessed via an API for user analysis or used internally by *CHYMP* for future integrations.

CHYMP also supports versatile scheduling options to meet steering, switching, and splitting requirements:

- Single-interface transmission, allowing users to select a preferred interface.
- Single-interface transmission with a backup interface, ensuring continuity in case of complete failure.
- QoS-aware single-interface transmission with a backup, switching paths as needed for compliance.
- Reliable transfer through packet duplication to enhance reliability.
- Throughput increase by aggregating capacity across multiple paths.

In summary, *CHYMP* enables comprehensive path management – identifying, selecting, adding, and removing paths –and supports advanced scheduling policies, effectively addressing QoS with packet reordering, duplicate removal, and critical metrics like packet loss, delay, and throughput.

V. Testbed Definition

The testbed consists of a client and server connected via a router to separate Wi-Fi and Ethernet interfaces, with a switch to simulate network failures.

On these machines, a prototype implementation of our *MPF-Gateway* generates network traffic by transmitting a text-file, which provides consistency across all tests.

A. Prototype implementation

The proof-of-concept was implemented in Go (Golang) version 1.22.04, utilizing the "golang.org/x/sys" package version 0.22.0 to interface with the underlying Ubuntu 22.04.4 operating system.

Therefore all packet processing in this implementation is done on the CPU, without any hardware acceleration and optimizations that would be used for real-world solutions.

B. Hardware Setup

- **Client:** A PC with an Intel Core i7-7700HQ CPU (4 cores, 8 threads, 2.80 GHz) is used as the client. It is connected via two interfaces: a 100 Mbps Ethernet link and an 80 Mbps wireless link, both 64-bit wide with a 33 MHz clock.
- **Server:** The server is a PC equipped with an Intel Core i5-2320 CPU (4 cores, 4 threads, 3.00 GHz), connected via a 100 Mbps Ethernet link with a 32-bit width and a 33 MHz clock.

For the testing of the aggregated network links, the single-link transmission was limited to 20 Mbps, to avoid running into any hardware limitations.

C. MPF Test Setup

The MPF package operates as an autonomous system, requiring only configuration files for initialization. It has been designed to support the functionalities described in Section III, supporting *CHYMP* and MPTCP.

The *MPF-Gateway* follows a client-server approach, as suggested in [12]. The client is responsible for most actions and decision-making, while the server primarily provides support.

The following tests were conducted to evaluate the *MPF-Gateway*'s functionality:

- **Control Protocol:** Validation of MPF messaging for connection setup, path management, and measurements.
- **Policy-Based Transmission:** Traffic tests to analyze the effect of different policies on data transmission.
- **Backup and Switching:** Disruption of an interface to test switching and system resilience.
- **QoS-Aware Transmission:** Testing system response to QoS violations by switching paths based on delay, throughput, and packet loss.
- **Throughput and Duplication:** Verification of throughput aggregation and reliability under packet loss using both interfaces.

D. MPTCP Test Setup

As a comparison for CHYMP, we chose MPTCP as a baseline, due to its widespread use in current networking systems. We evalueated MPTCP on:

- **Throughput and Transmission:** Evaluation of MPTCP throughput and transmission times.
- **Failure Handling:** Analysis of MPTCP's response to interface failure.

ISBN: 978-3-903176-71-3

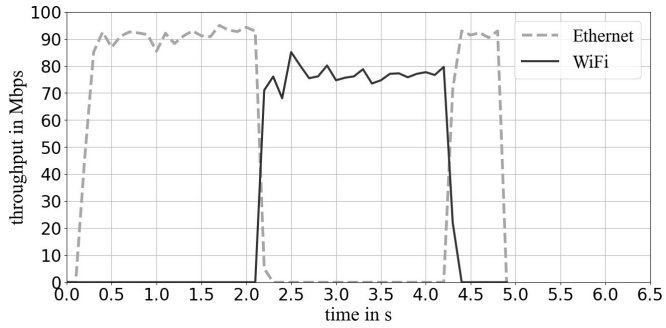

Figure 5. Interface switching due to interruption

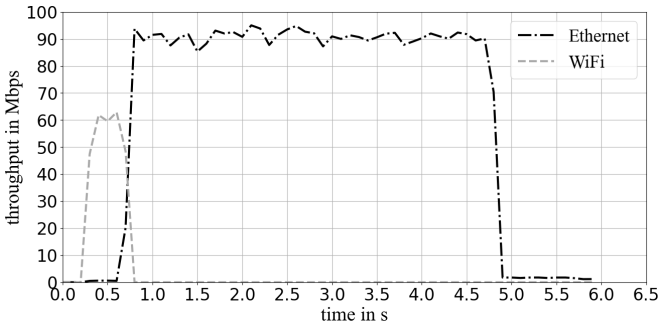

Figure 6. Interface switching for QoS compliance

VI. Test Results & Analysis

This chapter presents the test results obtained from the experimental setup in Section V, alongside analysis and discussion. The chapter is organized into two main sections covering the MPF system results and the MPTCP results, followed by a comparative evaluation of MPTCP and *CHYMP*.

A. MPF System

The interface measurement process evaluates maximum throughput, average delay, and packet loss for each interface using UDP traffic. In this setup, the MPF integrates a performance measurement function that continuously monitors these parameters during system initialization and upon detecting new interfaces. The gathered results, shown in Table I, reveal that the system accurately identifies maximum throughput under current network conditions.

For instance, the Ethernet interface initially showed a throughput of 93.61 Mbps, but packet loss indicated that 93.37 Mbps is the maximum achievable within an "Error-Free" transmission threshold.

We attribute the rather unusual Wi-Fi results to the kernel control of the packets, which limits the send rate to avoid errors. Additionally, the Wi-Fi 6 protocol's error correction mechanisms, which aim to reduce the impact of issues inthe physical layer, and the minimal interference in the test environment, explain the absence of packet loss. The low average delay values are attributed to the specific testbed configuration, with direct connections and spatial proximity.

Table I
Network Performance Metrics

Interface	Avg. Throughput	Avg. Delay	Packet Loss
Ethernet	93.61 Mbps	0.15 ms	0.25 %
Wi-Fi	74.45 Mbps	0.19 ms	0.00 %

In Figure 5, the switching behavior of the MPF system is illustrated for the fallback scenario without QoS requirements. Here, when the Ethernet interface fails, the Wi-Fi interface seamlessly takes over until the Ethernet connection stabilizes. This demonstrates the system's capability for interface failover, ensuring consistent connectivity.

Figure 6 depicts switching behavior prompted by QoS non-compliance due to a delay issue. In this scenario, a policy enforcing a 1 ms delay threshold for a specific client triggers the switch. The corresponding policy is defined in Figure 7.

```
{
    "policy_num": "UC03",
    "allowed_interfaces": [
        {
            "type": "Ethernet",
            "priority": 2,
            "name": "enp3s0f1",
            "ip": "192.168.1.12",
            "metrics": {
                "delayavg": 0.149524,
                "packetloss": 0,
                "throughput": 93
            }
        },
        {
            "type": "Wi-Fi",
            "priority": 1,
            "name": "wlp4s0",
            "ip": "192.168.0.242",
            "metrics": {
                "delayavg": 0.207738,
                "packetloss": 0,
                "throughput": 67
            }
        }
    ],
    "traffic_requirements": {
        "delay": 1,
        "throughput": 5,
        "packetloss": 1
    }
}
```

Figure 7. *CHYMP* policy specifying network traffic requirements

For this test, Wi-Fi was set as Priority 1 and Ethernet as Priority 2, highlighting flexible policy configurations.

The system associates each interface with its Network Interface Card (NIC) name, IP address, and initial metrics, updating values over time based on traffic. As induced latency increased Wi-Fi delay to 1.5 ms, exceeding the 1 ms threshold, the system switches to Ethernet.

ISBN: 978-3-903176-71-3

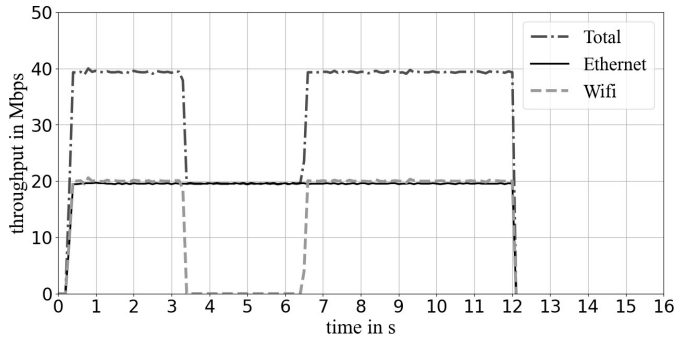

Figure 8. MPF link aggregation with interface failure

To test reliability, traffic is sent concurrently over both interfaces, each with independent packet error rates. The combined error rate is calculated to ensure it is lower than the highest individual error rate.

By adjusting packet error rates, we can confirm that the total error probability follows Equation (1):

$$Pe(A \cap B) = Pe(A) \cdot Pe(B) \quad (1)$$

This demonstrates the resilience of multi-interface scheduling, albeit with reduced throughput.

Figure 8 shows aggregated throughput results, confirming that combining both interfaces maximizes the data rate. Even with interface failure, transmission is sustained, highlighting the robustness of aggregation for high availability, unlike failover, which switches between links. Additionally, the simplified Layer 4 protocol omits congestion control, keeping transmission stable at maximum rates, with only the MPF autonomously adjusting throughput thresholds.

B. *MPTCP*

Figure 9 shows MPTCP transmission under failure conditions similar to Figure 8, where only the Wi-Fi interface fails. This test demonstrates that MPTCP halts transmission across all links temporarily, then resumes on the functional link after a delay (80 ms in this case), and eventually restores transmission on both links.

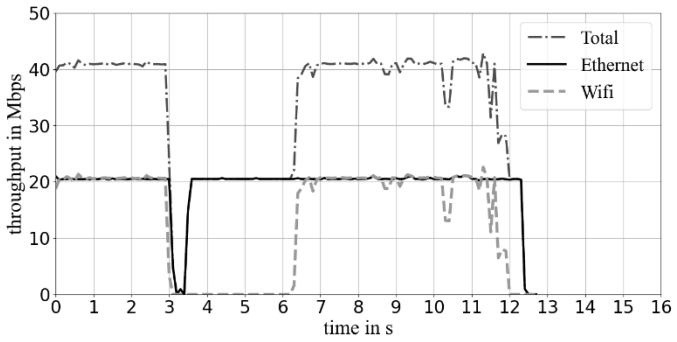

Figure 9. MPTCP link aggregation with interface failure

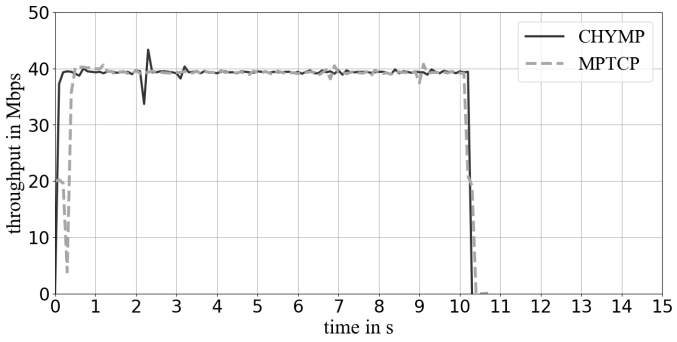

Figure 10. MPTCP vs. *CHYMP* performance comparison under stable conditions

This behavior reflects MPTCP's congestion control and fully-ordered delivery requirements, which address out-of-order packet delivery after a failure.

Throughput oscillations on the Wi-Fi link highlight how TCP congestion control agressively responds to to link disturbances or delays caused by packet retransmissions.

C. *MPTCP vs* CHYMP

While both protocols perform similarly in stable conditions, as shown in Figure 10, *CHYMP* achieves 5.5% higher effective throughput due to lower protocol overhead. Conversely, MPTCP establishes connections about 150% faster than *CHYMP*, offering an advantage in short, low-duration transmissions.

In Figure 11, both protocols are tested under simulated interface failure to observe transmission stability. Here, *CHYMP* completes the transfer 8.2% faster than MPTCP, showcasing its resilience in unstable networks.

This stability highlights *CHYMP*'s advantage in handling interface failures. Both protocols transmitted without packet loss, but MPTCP retains an edge in connection setup time, making it preferable for low-traffic or latency-sensitive use cases. In our tests, however, *CHYMP*'s setup delay becomes negligible after the third packet, minimizing the impact on sustained transfers.

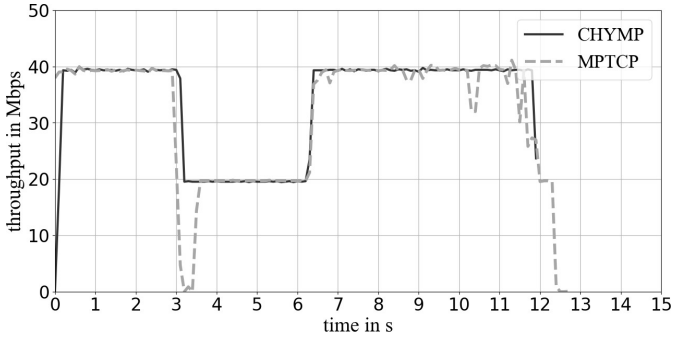

Figure 11. MPTCP vs. *CHYMP* performance comparison in unstable networks

ISBN: 978-3-903176-71-3

VII. Conclusion

In an era where the demand for efficient and reliable communication systems is continually increasing, this study has sought to address the limitations of existing multipath transport protocols. By developing an *MPF-Gateway* and our own protocol – *CHYMP*, we have aimed to enhance Quality of Service (QoS) in heterogeneous network environments.

The findings of this research not only demonstrate significant performance improvements but also lay the groundwork for future developments in multipath transport solutions. With this context in mind, we present the conclusions drawn from our integration and comparison with MPTCP.

A. CHYMP

When evaluated against MPTCP, our *CHYMP* protocol shows a distinct advantage in performance efficiency and reliability, especially under network instability.

Comparative results indicate that, while both MPTCP and *CHYMP* perform similarly in stable conditions, *CHYMP* achieves a 5.5% higher effective throughput with lower protocol overhead.

In unstable network scenarios, *CHYMP* completed transmissions 8.2% faster than MPTCP due to its robustness against interface failure, making it well-suited for applications in volatile network environments.

Though *CHYMP* has a slightly slower connection establishment time (0.9ms vs. 0.3ms for MPTCP), this delay is negligible in high-volume transfers, where its faster transmission speed provides significant gains.

B. MPF-Gateway

This paper presented a comprehensive evaluation of our Gateway solution, showcasing its capabilities in adaptive interface switching, high-availability link aggregation, and resilient multi-interface scheduling. The Gateway solution dynamically measures network performance, triggering seamless interface switching based on quality of service (QoS) policies.

Test results demonstrate that our system effectively detects and utilizes maximum throughput for each interface, ensuring reliable connectivity even during network instability. As shown in our testing, a policy violation due to elevated delay on Wi-Fi prompted a swift switch to Ethernet, underscoring the Gateway's responsiveness in maintaining QoS compliance.

The Gateway's failover mechanism allows uninterrupted connectivity by switching interfaces upon failure, and its link aggregation approach maximizes throughput by combining available bandwidth across interfaces. Furthermore, concurrent multi-interface scheduling enhances resilience, as verified by our probabilistic analysis, which matches observed packet error rates across interfaces.

Our solution's capability to switch proxy operating modes, selecting different protocols for translation, ensures adaptability to emerging networking technologies. This flexibility allows specific traffic to be routed through chosen protocols, leveraging each protocol's unique strengths.

Overall, our Gateway solution demonstrates significant advantages in network resilience, efficiency, and responsiveness compared to baseline MPTCP. By leveraging the strengths of both interface selection and protocol efficiency, the system offers a scalable and reliable approach to network management in multi-interface environments, suitable for applications requiring both high throughput and seamless connectivity while avoiding the modification of end-user applications, and instead, adding middleboxes to the connection points of users and servers.

References

[1] F. A. W. Group, *Future Railway Mobile Communication System: SystemRequirements Specification*, version 1.0.0, Accessed: 2024-10-29, Feb. 2023.

[2] J. Beeharry and B. Nowbutsing, "Forecasting IPv4 Exhaustion and IPv6 Migration," in *2016 IEEE International Conference on Emerging Technologies and Innovative Business Practices for the Transformation of Societies (EmergiTech)*, IEEE, 2016, pp. 336–340.

[3] J. Iyengar and M. Thomson, *QUIC: A UDP-Based Multiplexed and Secure Transport*, RFC 9000, May 2021. [Online]. Available: https://www.rfc-editor.org/info/rfc9000.

[4] T. Pauly, E. Kinnear, and D. Schinazi, *An Unreliable Datagram Extension to QUIC*, RFC 9221, Mar. 2022. [Online]. Available: https://www.rfc-editor.org/info/rfc9221.

[5] R. R. Stewart, M. Tüxen, and karen Nielsen, *Stream Control Transmission Protocol*, RFC 9260, Jun. 2022. [Online]. Available: https://www.rfc-editor.org/info/rfc9260.

[6] 3GPP, "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; System architecture for the 5G System (5GS); Stage 2 (Release 18)," 3GPP, Tech. Rep. TS 23.501 V18.5.0 (2024-03), 2024.

[7] Y. Liu, Y. Ma, Q. D. Coninck, O. Bonaventure, C. Huitema, and M. Kühlewind, "Multipath Extension for QUIC," Internet Engineering Task Force, Internet-Draft draft-ietf-quic-multipath-11, Oct. 2024, Work in Progress, 38 pp. [Online]. Available: https://datatracker.ietf.org/doc/draft-ietf-quic-multipath/11/.

[8] J. Han, K. Xue, W. Wei, Y. Xing, J. Liu, and P. Hong, "Transparent Multipath: Using Double MPTCP Proxies to Enhance Transport Performance for Traditional TCP," *IEEE Network*, vol. 35, no. 5, pp. 181–187, 2021.

[9] Y. Kojima, T. Kawasaki, J. Suga, and R. Takechi, "A Method of Introducing Multipath TCP to Mobile Core Networks and Its Evaluations," in *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*, 2016, pp. 1–5.

[10] 3. G. P. Project, "Access Traffic Steering, Switching and Splitting (ATSSS); Stage 3," Technical Specification Group Core Network and Terminals, Technical Specification V19.0.0, Sep. 2024, Release 19. [Online]. Available: https://portal.3gpp.org/docbase/Document_Archive/24_series/24.193/V19.0.0-Release-19.

[11] S. Kanugovi, F. Baboescu, J. Zhu, J. Mueller, and S. Seo, *Multiple Access Management Services Multi-Access Management Services (MAMS)*, RFC 8743, Mar. 2020. [Online]. Available: https://www.rfc-editor.org/info/ rfc8743.

[12] 3GPP, "3rd generation partnership project; technical specification groupcore network and terminals; 5g system; access traffic steering, switchingand splitting (atsss); stage 3 (release 18)," 3GPP, Tech. Rep. TS 24.193 V18.5.0 (2024-03), 2024.

ISBN: 978-3-903176-71-3