

Lab WiFi Report

Andrea Botticella*

andrea.botticella@studenti.polito.it
Politecnico di Torino
Turin, Italy

Renato Mignone*

renato.mignone@studenti.polito.it
Politecnico di Torino
Turin, Italy

Elia Innocenti*

elia.innocenti@studenti.polito.it
Politecnico di Torino
Turin, Italy

Simone Romano*

simone.romano2@studenti.polito.it
Politecnico di Torino
Turin, Italy

ABSTRACT

This report investigates the performance of wireless and device-to-device communication by comparing theoretical predictions with experimental results in three distinct scenarios: both devices connected via WiFi, both via Ethernet, and a mixed configuration where one device uses Ethernet and the other WiFi. Using tools such as iperf3 and Wireshark, we measured the goodput and analyzed its variability under different network conditions. The experimental results are compared with theoretical calculations based on protocol efficiencies and network overheads. Our findings highlight the stability of Ethernet connections and the challenges posed by WiFi's shared medium and half-duplex constraints. This study provides useful insights for optimizing network configurations in mixed environments.

1 BACKGROUND AND OBJECTIVES

1 In this laboratory exercise, we aim to evaluate and compare the
2 performance of wired and wireless communication within a local
3 area network. The experiment involves setting up three distinct
4 scenarios: both devices connected via WiFi, both devices connected
5 via Ethernet, and a mixed configuration where one device uses
6 Ethernet and the other uses WiFi.
7 The main objectives of the lab are to:

- 8 • Measure the goodput, i.e., the rate of useful data received at the
9 application layer, using iperf3.
- 10 • Analyze the variability and stability of the connection in each
11 scenario by collecting data over multiple test runs.
- 12 • Compare the experimental results against theoretical predictions
13 based on protocol efficiencies and network overhead.
- 14 • Investigate potential sources of performance degradation in wire-
15 less communication, such as interference, half-duplex operation,
16 and shared medium limitations.

17 Through these experiments, we will gain practical insights into
18 the strengths and limitations of both Ethernet and WiFi communi-
19 cation methods, which are crucial for optimizing network perfor-
20 mance in mixed connectivity environments.

2 METHODOLOGY AND CONCEPTS

21 This section outlines the experimental setup, the tools employed for
22 the measurements, and the theoretical basis for estimating goodput.

*The authors collaborated closely in developing this project.

2.1 Selected Tools

To evaluate the performance of both Ethernet and WiFi connections, we utilized several specialized tools:

- **iperf3**: Used to generate traffic and measure goodput in both TCP and UDP modes. By executing repeated tests, iperf3 provides key metrics such as minimum, maximum, average, and standard deviation of the throughput.
- **Wireshark**: Employed to capture and analyze network traffic, Wireshark enabled us to inspect data flows, identify control and data frames, and validate experimental results.
- **Automation Script**: A Python script was developed to automate the entire measurement process. This script manages both server and client modes of iperf3, logs output in JSON, CSV, and plain text formats, and computes summary statistics. The script accepts several command-line flags:
 - **-server**: Launches the iperf3 server in JSON output mode.
 - **<SERVER_IP>**: Specifies the server's IP address when running in client mode.
 - **-udp**: Switches the test from the default TCP mode to UDP.
 - **-bitrate**: Sets the target bitrate for UDP tests (e.g., 10M for 10 Mbps).
 - **-iter**: Determines the number of test iterations to perform. The output files (logs, JSON, CSV reports, and raw output) generated by this script are used to document the experimental results and facilitate further analysis.

2.2 Goodput Estimation

Goodput represents the rate at which useful data is delivered to the application layer, excluding protocol overheads and retransmitted packets. The theoretical estimation of goodput is based on the efficiency of the protocol and the capacity of the network link:

$$G \leq \eta_{\text{protocol}} \times C,$$

where C is the capacity of the bottleneck link and η_{protocol} is the protocol efficiency.

For Ethernet connections, the efficiency for TCP is computed as:

$$\eta_{TCP}^{Eth} = \frac{MSS}{MSS + \text{TCP headers} + \text{IP headers} + \text{Ethernet overhead}}$$

with the Maximum Segment Size (MSS) defined as the MTU minus the headers. For a standard MTU of 1500 bytes, we obtain:

- $MSS \approx 1460$ bytes (after subtracting 20 bytes for the IP header and 20 bytes for the TCP header),
- An additional Ethernet overhead of approximately 38 bytes.

61 Thus, the efficiency for TCP over Ethernet is approximately:

$$\eta_{TCP}^{Eth} \approx \frac{1460}{1460 + 20 + 20 + 38} \approx 94.9\%.$$

62 Similarly, the efficiency for UDP is computed as follows. Since UDP
63 has an 8-byte header, its MSS is given by:

- 64 • MSS ≈ 1472 bytes (after subtracting 20 bytes for the IP header
65 and 8 bytes for the UDP header).

66 Thus, the efficiency for UDP over Ethernet is given by:

$$\eta_{UDP}^{Eth} \approx \frac{1472}{1472 + 20 + 8 + 38} \approx 95.7\%.$$

67 For WiFi, additional factors must be considered due to its half-
68 duplex nature and the inherent overhead of the 802.11 protocol (e.g.,
69 control frames, retransmissions, and channel contention). Conse-
70 quently, the effective efficiency is reduced by a WiFi-specific factor
71 (η_{WiFi}). The adjusted efficiency for TCP over WiFi can be expressed
72 as:

$$\eta_{TCP}^{WiFi} = \eta_{TCP}^{Eth} \times \eta_{WiFi},$$

73 with η_{WiFi} typically around 80% in optimal conditions. A similar
74 adjustment applies for UDP.

75 These theoretical estimates set an upper bound on the achiev-
76 able goodput, against which our experimental results are compared.
77 Discrepancies between theory and practice are primarily due to
78 dynamic environmental factors, such as interference, channel vari-
79 ability, and the inherent limitations of wireless communication.

3 EXPERIMENTAL SETUP AND TEST CASES

3.1 Equipment and Configuration

80 In this section, we describe the hardware and software configura-
81 tion used to perform our network performance measurements.
82 Table 1 summarizes the main devices, their interfaces, and relevant
83 specifications.

Device	Key Specifications
PC1	Victus 16-s1005nl Notebook <i>Operating System:</i> Ubuntu 24.04.2 LTS <i>Ethernet Interface:</i> Realtek RTL8111/8168/8211/8411 <i>Wireless Interface:</i> Realtek RTL8852BE (802.11ax) 2x2
PC2	Microsoft Surface Laptop Go 3 <i>Operating System:</i> Ubuntu 24.10 <i>Ethernet Interface:</i> via Anker PowerExpand+ USB-C Hub <i>Wireless Interface:</i> Intel Alder Lake-P CNVi (802.11ax) 2x2
Router	Vodafone Power Station Wi-Fi 6 <i>Ethernet Ports:</i> 4 × 1 GbE ports <i>Wi-Fi:</i> Dual-band 802.11ax (2.4 GHz 2x2, 5 GHz 4x4)
Cables	CAT.5E (up to 1 Gbps)

Table 1: Summary of Hardware and Network Configuration

85 This hardware setup allows us to compare Ethernet versus WiFi
86 performance under a consistent router and cabling environment.
87 In the next section, we detail the evaluation scenarios and the
88 measurement methodology.

Connections	Key Specifications		
Ethernet	<i>Cabling:</i> CAT.5E <i>Nominal Speed:</i> 1 Gbps	<i>Protocol:</i> 802.3??	
WiFi	<i>Standard:</i> 802.11ax <i>Nominal Speed:</i> 1200 Mbps <i>Bandwidth:</i> 80 MHz	<i>Security Protocol:</i> ? <i>Frequency:</i> 5 GHz <i>Channel:</i> ?	

Table 2: Ethernet and WiFi Connection Specifications

3.2 Evaluation Scenarios

We considered three distinct network configurations to assess the performance differences between wired and wireless communications. For each scenario, the theoretical goodput is computed based on the nominal link capacity and protocol efficiency.

1. Both Ethernet:

In this scenario, both PC1 and PC2 are connected to the router via CAT.5E cables, providing a nominal link capacity of 1 Gbps. The efficiency for Ethernet is calculated as follows:

$$\eta_{TCP}^{Eth} \approx \frac{1460}{1460 + 20 + 20 + 38} \approx 94.9\%,$$

$$\eta_{UDP}^{Eth} \approx \frac{1472}{1472 + 20 + 8 + 38} \approx 95.7\%.$$

Thus, the expected goodput is:

$$G_{TCP}^{Eth} \leq 0.949 \times 1000 \text{ Mbps} \approx 949 \text{ Mbps},$$

$$G_{UDP}^{Eth} \leq 0.957 \times 1000 \text{ Mbps} \approx 957 \text{ Mbps}.$$

2. Both WiFi:

For this configuration, both devices use their wireless interfaces (802.11ax) to connect to the router. Although the nominal WiFi link speed is assumed to be approximately 867 Mbps, the half-duplex nature of WiFi effectively halves the throughput available for data transfer. Assuming a WiFi efficiency factor of about 80%, the expected goodput for TCP is:

$$G_{TCP}^{WiFi} \leq 0.80 \times 867 \text{ Mbps} \times \frac{1}{2} \approx 347 \text{ Mbps},$$

and similarly for UDP:

$$G_{UDP}^{WiFi} \leq 0.806 \times 867 \text{ Mbps} \times \frac{1}{2} \approx 350 \text{ Mbps}.$$

3. Mixed Scenario:

In this configuration, one device (PC1) is connected via Ethernet while the other (PC2) uses its WiFi interface. Here, the bottleneck is the WiFi link; however, since only one device is utilizing WiFi, the throughput is not halved. The expected goodput is then:

$$G_{TCP}^{Mixed} \leq 0.80 \times 867 \text{ Mbps} \approx 694 \text{ Mbps},$$

$$G_{UDP}^{Mixed} \leq 0.806 \times 867 \text{ Mbps} \approx 698 \text{ Mbps}.$$

These calculations provide the theoretical upper bounds for goodput in each scenario. The experimental results, obtained via automated measurements using the provided Python script, are compared against these predictions to evaluate real-world performance.

Note on Experimental Automation:

The experimental process was automated using a custom Python

script. This script manages both the server and client modes of iperf3 and handles output logging in JSON, CSV, and plain text formats. Key command-line flags include:

- **-server**: Runs the iperf3 server in JSON output mode.
- **<SERVER_IP>**: Specifies the server's IP address for client mode.
- **-udp**: Switches the test from TCP to UDP.
- **-bitrate**: Sets the target bitrate for UDP tests (e.g., 10M).
- **-iter**: Specifies the number of test iterations.

The output files generated by the script serve as the basis for our statistical analysis and subsequent comparison with the theoretical predictions.

4 ANALYSIS AND FINDINGS

4.1 TCP Performance

The performance tests using TCP reveal several noteworthy trends. The analysis for TCP performance in different scenarios is organized as follows:

1. Both Ethernet:

Figure 1 shows the TCP throughput measured in the Ethernet scenario. The graph reveals a rapid ramp-up in throughput during the first few seconds, followed by a stable transmission rate that approaches the theoretical value.

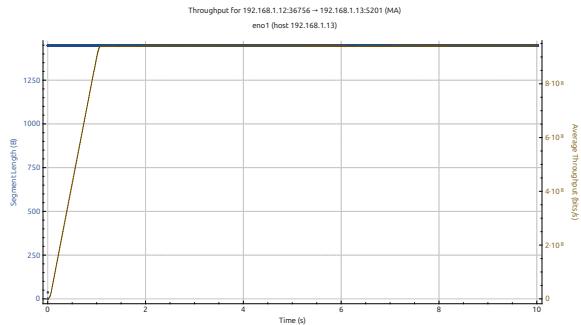


Figure 1: TCP Throughput in the Ethernet Scenario.

Figure 2 illustrates the round-trip time (RTT), which remains very low (typically within a few milliseconds), highlighting the minimal latency in wired connections.

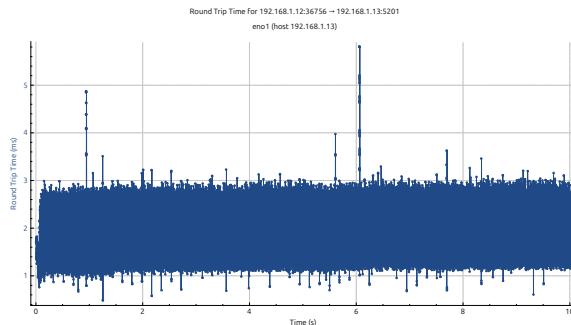


Figure 2: TCP Round Trip Time in the Ethernet Scenario.

Furthermore, the I-O graph (Fig. 3) confirms a consistent packet flow with little variation, indicating that the Ethernet setup effectively utilizes the available capacity.

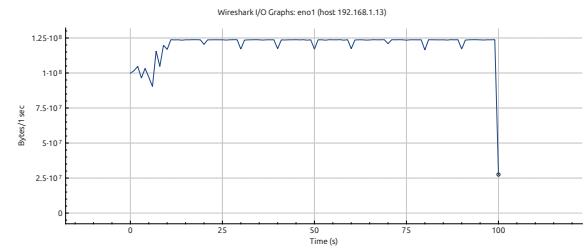


Figure 3: Wireshark I-O Graph for TCP in the Ethernet Scenario.

Overall, the Ethernet scenario demonstrates a near-ideal performance with high throughput and minimal latency, closely matching the theoretical predictions.

2. Both WiFi:

In the WiFi scenario, the throughput graph (Fig. 4) shows an initial ramp-up phase during the first 2 seconds, after which the throughput fluctuates around an average value that is significantly lower than the theoretical maximum of approximately 347 Mbps. These fluctuations suggest that protocol overhead, wireless interference, and the half-duplex nature of WiFi adversely affect performance.

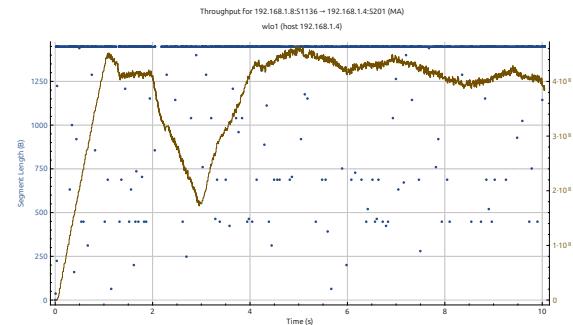


Figure 4: TCP Throughput in the WiFi Scenario.

The round-trip time (RTT) measurements (Fig. 5) reveal RTT values ranging from about 50 to 200 ms, indicating intermittent delays likely due to congestion and contention in the wireless medium.

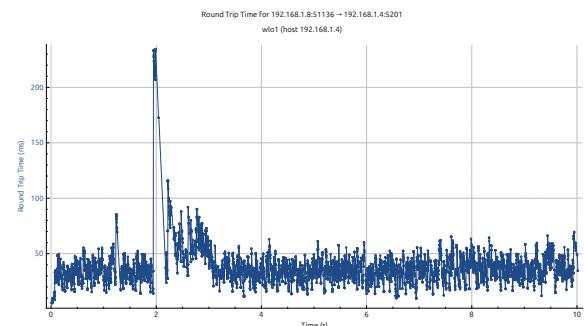


Figure 5: TCP Round Trip Time in the WiFi Scenario.

Furthermore, the I-O graph (Fig. 6) illustrates a variable number of transmitted packets per interval, reflecting the dynamic

167 nature of WiFi communication where channel conditions and
 168 collision avoidance mechanisms influence performance.

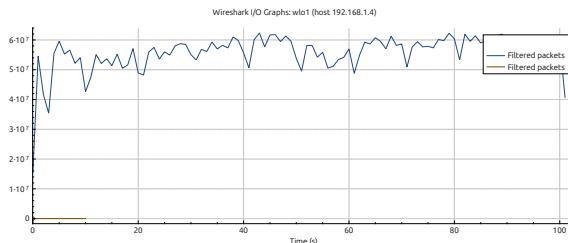


Figure 6: Wireshark I-O Graph for TCP in the WiFi Scenario.

169 Overall, while the theoretical capacity for TCP over WiFi is
 170 estimated to be around 347 Mbps, the experimental data indicate
 171 that real-world factors substantially reduce the effective
 172 throughput.

3. Mixed:

173 Figure 7 displays the TCP throughput for the mixed configuration.
 174 The graph shows that the throughput reaches a stable level
 175 after an initial ramp-up phase, although it remains below the
 176 Ethernet scenario and is consistent with the expected reduction
 177 due to the reliance on the wireless link.
 178
 179

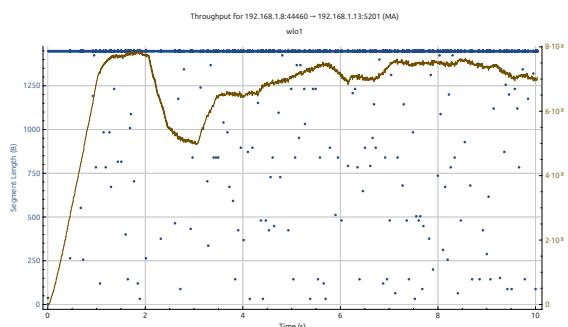


Figure 7: TCP Throughput in the Mixed Ethernet/WiFi Scenario.

180 The round-trip time (RTT) measurements, presented in Figure 8,
 181 indicate moderate latency, with RTT values generally remaining
 182 within a lower range compared to the pure WiFi scenario.
 183 This suggests that the wired segment helps in reducing overall
 184 latency.
 185

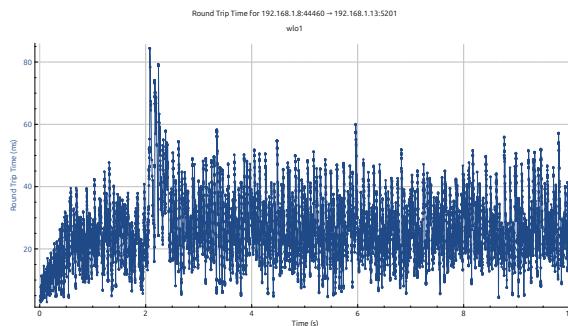


Figure 8: TCP Round Trip Time in the Mixed Ethernet/WiFi Scenario.

186 The I-O graph for TCP (Fig. 9) shows a relatively steady packet
 187 flow over the test intervals, confirming that the mixed configura-
 188 tion maintains a stable performance despite the inherent
 189 variability of the wireless link.

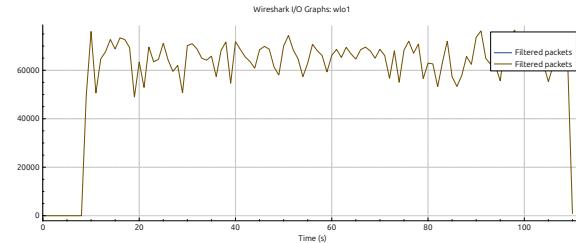


Figure 9: Wireshark I-O Graph for TCP in the Mixed Scenario.

4. Shared Capacity:

190 In this scenario, a third host connected to the same access point
 191 was concurrently downloading the film "Natale a Rio" (directed
 192 by Neri Parenti), which introduced significant interference dur-
 193 ing the tests. This additional traffic compromised the available
 194 network capacity, leading to degraded performance. Figure 10
 195 shows the TCP throughput under this shared capacity condi-
 196 tion. Compared to the mixed scenario without interference, the
 197 throughput exhibits a notable decrease. The average throughput
 198 is lower, reflecting the reduced available bandwidth caused by
 199 the competing download traffic.
 200

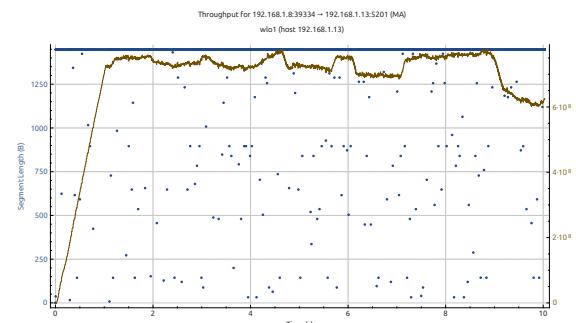


Figure 10: TCP Throughput in the Shared Capacity Scenario (with third-host interference).

201 The round-trip time measurements (Fig. 11) indicate increased
 202 variability and slightly elevated latency. Although the RTT
 203 values remain relatively moderate, the fluctuations suggest that
 204 the network experiences occasional congestion and delays as a
 205 result of the third host's activity.

206 The I-O graph for TCP (Fig. 12) further confirms the impact of
 207 the interference. The graph displays irregular intervals and a
 208 lower packet transmission rate compared to the mixed scenario
 209 without the additional load, demonstrating how the extra traffic
 210 disrupts the steady flow of data.

4.2 UDP Performance

211 The UDP tests offer an insightful comparison to the TCP results
 212 by eliminating congestion control and acknowledgment overhead.
 213 The analysis for UDP performance across different scenarios is
 214 structured as follows:
 215

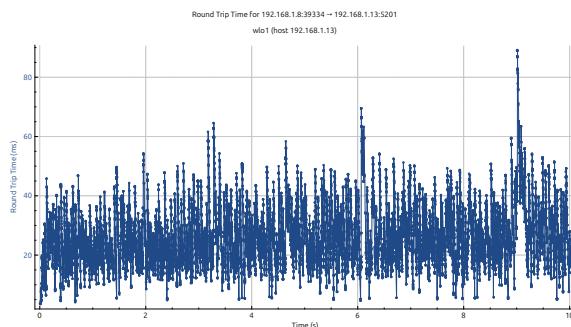


Figure 11: TCP Round Trip Time in the Shared Capacity Scenario.

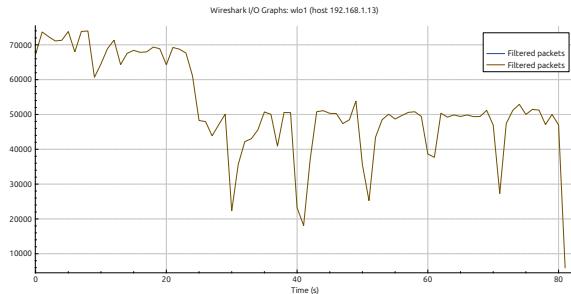


Figure 12: Wireshark I-O Graph for TCP in the Shared Capacity Scenario.

216 1. Both Ethernet:

217 In the Ethernet configuration, the I-O graph for UDP (Fig. 13)
218 indicates a steady packet flow, with minimal fluctuations compared
219 to the WiFi scenario.

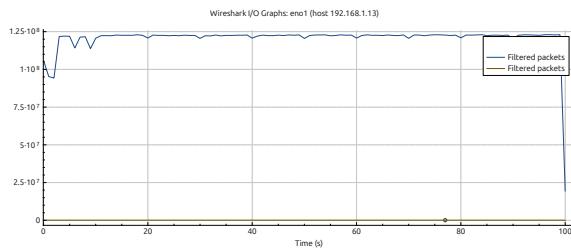


Figure 13: Wireshark I-O Graph for UDP in the Ethernet Scenario.

220 The throughput achieved is very close to the theoretical prediction,
221 confirming that the wired setup reliably supports high-speed data transfer.
222 The absence of retransmission or congestion control overhead in UDP further contributes to this consistent performance.

223 In summary, the wired (Ethernet) tests demonstrate that both
224 TCP and UDP protocols achieve performance levels very close
225 to their theoretical capacities, with TCP showing stable through-
226 put and low latency, and UDP exhibiting a consistent packet flow
227 and high throughput. This confirms that, in a controlled wired
228 environment, network performance is minimally impacted by
229 protocol overhead or environmental factors.

230 2. Both WiFi:

231 In the WiFi scenario, the I-O graph for UDP (Fig. 14) demon-
232 strates a more consistent packet flow compared to TCP. How-
233 ever, despite the smoother transmission, the overall throughput

234 remains below the theoretical upper bound. The lack of retruns-
235 mission mechanisms in UDP allows for slightly higher instant-
236 taneous throughput; nonetheless, factors like interference and
237 channel contention continue to impact performance.

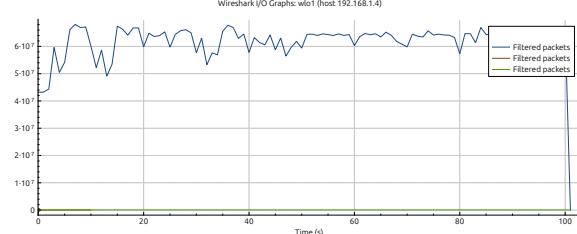


Figure 14: Wireshark I-O Graph for UDP in the WiFi Scenario.

238 A direct comparison between TCP and UDP in the WiFi scenario
239 shows that UDP can achieve marginally higher throughput due
240 to its reduced overhead. Nonetheless, both protocols suffer from
241 real-world limitations that prevent them from reaching their
242 theoretical capacities. This discrepancy between capacity and
243 theoretical goodput emphasizes the impact of wireless inter-
244 ference, channel contention, and protocol-specific overhead on
245 performance.

246 3. Mixed:

247 In the mixed scenario, the UDP I-O graph (Fig. 15) indicates
248 a consistent flow of packets, similar to the TCP case but with
249 slightly less variability due to the absence of congestion control.

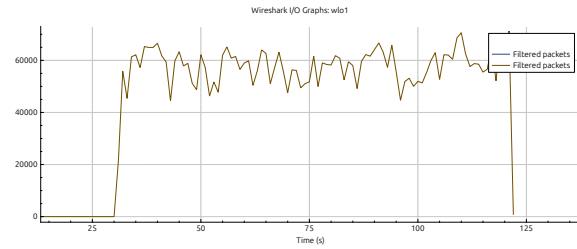


Figure 15: Wireshark I-O Graph for UDP in the Mixed Scenario.

250 The throughput observed is in line with expectations given that
251 only the wireless link acts as the bottleneck.

252 In summary, the mixed scenario demonstrates that while the
253 presence of a wired connection on one end improves overall
254 latency and stability compared to a full WiFi configuration,
255 the performance remains primarily constrained by the wireless
256 link. Both TCP and UDP protocols achieve throughput values
257 that are consistent with theoretical predictions for a mixed
258 Ethernet/WiFi environment.

259 4. Shared Capacity:

260 The UDP tests under the shared capacity scenario also reveal the
261 negative impact of the additional download traffic. Although
262 UDP is less affected by protocol overhead, the increased con-
263 tention for the wireless medium leads to degraded performance.
264 The UDP performance in this scenario is indirectly reflected in
265 the I-O graph (Fig. 16). The steady flow of packets observed in
266 a non-interfered environment is disrupted, resulting in a lower
267 effective throughput.

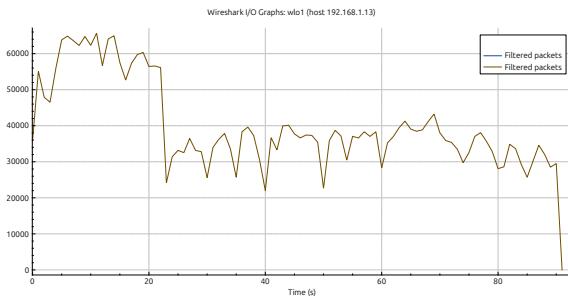


Figure 16: Wireshark I-O Graph for UDP in the Shared Capacity Scenario.

A APPENDIX

...

code here

Listing 1: Snipped name

312

313

271 Despite the inherent resilience of UDP to retransmission de-
272 lays, the interference from the third host causes a noticeable
273 reduction in performance. The graph shows that the packet flow
274 is not as consistent, further underlining the effects of shared
275 capacity when additional traffic is present.

276 Overall, the shared capacity scenario clearly demonstrates that
277 when a third host generates significant traffic (as in the case
278 of streaming a movie), the available network resources are fur-
279 ther divided, leading to performance degradation for both TCP
280 and UDP protocols. This scenario highlights the importance of
281 considering real-world usage patterns and interference when
282 designing and evaluating network performance.

5 CONCLUSION

283 In this project, we evaluated the performance of network commu-
284 nication under various scenarios using both wired (Ethernet) and
285 wireless (WiFi) connections. Our experimental results, obtained
286 through automated measurements with iperf3 and detailed packet
287 analysis with Wireshark, were compared against theoretical pre-
288 dictions of goodput for both TCP and UDP protocols.

289 Overall, the Ethernet scenario demonstrated near-ideal perfor-
290 mance, with throughput and latency closely matching the theoreti-
291 cal values. This confirms that a controlled wired environment can
292 efficiently utilize available bandwidth with minimal interference. In
293 contrast, the WiFi scenario showed a significant performance drop,
294 with fluctuations in throughput and increased latency due to the in-
295 herent limitations of wireless communication such as interference,
296 contention, and the half-duplex nature of WiFi.

297 The mixed scenario, where one device is connected via Ethernet
298 and the other via WiFi, presented an intermediate case. Here, while
299 the wired segment helped in reducing latency and stabilizing per-
300 formance, the overall throughput remained limited by the wireless
301 link. Finally, the shared capacity scenario—where an additional host
302 engaged in heavy traffic (streaming a movie)—further degraded per-
303 formance for both TCP and UDP tests. This clearly highlights the
304 impact of network congestion and shared medium contention on
305 real-world performance.

306 These findings emphasize the importance of considering environ-
307 mental and traffic-related factors when designing and optimizing
308 network infrastructures. While theoretical models provide useful
309 upper bounds, actual network performance is influenced by a range
310 of practical factors that must be taken into account for effective
311 network planning and troubleshooting.