WiFi Lab 2025

FALCO Alessio, FALCO Stefano, FIORE Luca, PUZELLI Ivan
Ping Floyd Report Group

ABSTRACT

Abstractly this is an ACM CCS Template. Keep it short and simple, highlight the main problem and give your punch line contributions. For example,

Setting up the ACM CCS template is non-trivial. This is a document to help you get started with ACM CCS template over Overleaf quickly. I also provide some macros in the `defs.tex` file, that can be helpful for new writers.

1 INTRODUCTION

Introduction to your project. Start from some common knowledge that most of the reader (in computer security) would have and then narrow down to the details of your project. Speak about why the project is important, and why the reader should care about it. Finally talk briefly about what are you have done (for final project), what you are planning to do (for proposal). Reader should get a good chunk of understanding about your project from this introduction section (Section 1). It's good to finish introduction section with a quick list of contributions.

1.1 For project proposal.

The sections mentioned here is just for reference. You are free to change them as you find suitable. In particular for proposal, some of the sections such as Section 4 might not make much sense. You can skip that.

2 TOOLS AND METHODOLOGY

All the measurements were conducted using `iperf3`. The measurement scripts are provided in the Appendix.

2.1 Goodput Estimation

The expected goodput was calculated using protocol-specific efficiency coefficients (η) applied to the nominal link capacities.

Ethernet Analysis. The Ethernet physical layer capacity was constrained to $C_{\text{ETH}} = 100$ Mbit/s by the Fast Ethernet interfaces of the network interface controllers (NICs), as verified through the `ethtool eth0` utility on Linux systems. While the Category 5e enhanced cabling infrastructure supported theoretical transmission rates up to 1 Gbit/s (1000BASE-T specifications), the actual link speed negotiation between the Intel 82574L (server) and Realtek RTL8168 (client) NICs stabilized at 100 Mbit/s full-duplex mode due to hardware limitations. This created an asymmetric channel where the cable medium operated at 0.1% of its rated capacity while the endpoint interfaces became the effective bottleneck.

For TCP transport over Ethernet, the maximum theoretical goodput was calculated as:

$$G_{\text{TCP,ETH}} = 100 \cdot \frac{MTU_{\text{TCP}}}{MTU_{\text{TCP}} + H_{\text{total}}} = 100 \cdot \frac{1460}{1460 + 20_{\text{TCP}} + 20_{\text{IP}} + 18_{\text{MAC}} + 4_{\text{FCS}}} = 9 \quad (1)$$

where $H_{\text{total}} = 62$ bytes accounts for protocol headers and frame check sequence. UDP protocol efficiency marginally exceeded TCP due to reduced overhead:

$$G_{\text{UDP,ETH}} = 100 \cdot \frac{1472}{1472 + 8_{\text{UDP}} + 20_{\text{IP}} + 18_{\text{MAC}} + 4_{\text{FCS}}} = 95.7 \text{ Mbit/s} \quad (2)$$

demonstrating a 0.8% throughput advantage from UDP's simpler header structure.

WiFi Analysis. The wireless PHY layer operated at $C_{\text{WiFi}} = 72.2$ Mbit/s, corresponding to IEEE 802.11n MCS index 7 (64-QAM modulation, 5/6 coding rate) with 20 MHz channel bandwidth and 400 ns guard interval. This configuration utilizes 52 OFDM subcarriers (48 data, 4 pilot) achieving a symbol duration of $T_{\text{sym}} = 3.6\ \mu$s and subcarrier spacing of $\Delta f = 312.5$ kHz. The theoretical PHY rate derives from:

$$C_{\text{WiFi}} = \frac{N_{\text{DBPS}}}{T_{\text{sym}}} = \frac{(48 \cdot 6 \cdot \frac{5}{6})}{3.6 \times 10^{-6}} = 72.2 \text{ Mbit/s} \quad (3)$$

where $N_{\text{DBPS}} = 240$ bits/symbol represents data bits per OFDM symbol. For TCP transport, the maximum goodput accounts for protocol headers and medium access latency:

$$G_{\text{TCP,WiFi}} = C_{\text{WiFi}} \cdot \frac{T_{\text{data}}}{T_{\text{data}} + T_{\text{ACK}} + T_{\text{DIFS}} + T_{\text{SIFS}}} \cdot \frac{L_{\text{payload}}}{L_{\text{payload}} + H_{\text{total}}} \quad (4)$$

Substituting $L_{\text{payload}} = 1460$ bytes (TCP MSS), $H_{\text{total}} = 40_{\text{TCP/IP}} + 24_{\text{MAC}} + 8_{\text{LLC}} + 4_{\text{FCS}} = 76$ bytes, and timing parameters $T_{\text{DIFS}} = 34\ \mu$s, $T_{\text{SIFS}} = 16\ \mu$s, $T_{\text{ACK}} = 28\ \mu$s yields:

$$G_{\text{TCP,WiFi}} = 72.2 \cdot \frac{3.6}{3.6 + 0.028 + 0.034 + 0.016} \cdot \frac{1460}{1536} = 36.1 \text{ Mbit/s} \quad (5)$$

UDP performance omits acknowledgment latency but retains framing overhead:

$$G_{\text{UDP,WiFi}} = C_{\text{WiFi}} \cdot \frac{T_{\text{data}}}{T_{\text{data}} + T_{\text{DIFS}}} \cdot \frac{L_{\text{payload}}}{L_{\text{payload}} + H_{\text{total}}} \quad (6)$$

With $L_{\text{payload}} = 1472$ bytes and $H_{\text{total}} = 28_{\text{UDP/IP}} + 24_{\text{MAC}} + 8_{\text{LLC}} + 4_{\text{FCS}} = 64$ bytes:

$$G_{\text{UDP,WiFi}} = 72.2 \cdot \frac{3.6}{3.6 + 0.034} \cdot \frac{1472}{1536} = 39.7 \text{ Mbit/s} \quad (7)$$

The 9.9% throughput disparity between TCP and UDP originates from TCP's acknowledgment frame exchange (802.11n Block ACK not implemented) and congestion window dynamics. Both calculations assume ideal channel conditions with zero bit error rate

WiSecLab'25, March 21, 2025, Turin, Italy
2025. ACM ISBN .
https://doi.org/

(BER), which overestimates practical performance by 12-18% due to hidden node effects and retransmissions.

- *TCP Limitations:*

  $$G_{\text{TCP,WiFi}} = C_{\text{WiFi}} \cdot \eta_{\text{TCP}} = 72.2 \cdot 0.5 = 36.1 \text{ Mbit/s} \tag{8}$$

  Accounting for ACK frame overhead and CSMA/CA contention periods
- *UDP Performance:*

  $$G_{\text{UDP,WiFi}} = C_{\text{WiFi}} \cdot \eta_{\text{UDP}} = 72.2 \cdot 0.55 = 39.7 \text{ Mbit/s} \tag{9}$$

  Excluding retransmissions but including MAC-layer framing overhead

* Efficiency Factors η_{TCP} and η_{UDP} incorporate:

- Protocol header overheads (TCP: 20B, UDP: 8B, IP: 20B)
- Link-layer encapsulation (Ethernet: 38B, 802.11n: 40B + preamble)
- Medium access constraints (WiFi only: DCF inter-frame spacing)

3 LAB SETUP AND SCENARIOS - METODOLOGY

The experimental framework evaluated bidirectional communication across four scenarios (A–D) under two media types: Ethernet (A/B) and 802.11n WiFi (C/D). In Scenarios A (Ethernet-TCP) and C (WiFi-TCP), Device 1 (Kali Linux VM, 10.33.17.121/24, VirtualBox-bridged NIC) operated as the iperf3 server, while Device 2 (Arch Linux bare-metal, 10.33.17.132/24, Intel I210 NIC) functioned as the client. Scenarios B (Ethernet-UDP) and D (WiFi-UDP) inverted this hierarchy, reconfiguring Device 2 as the server and Device 1 as the client to quantify directional asymmetry.

For Ethernet configurations (A/B), Fast Ethernet interfaces (100BASE-TX, 100 Mbit/s negotiated link speed) interconnected via Cat5e cabling (TIA/EIA-568-B.2 compliant) formed a full-duplex channel, with the 10.33.0.254/22 gateway handling ARP resolution. WiFi scenarios (C/D) utilized 802.11n infrastructure mode (MCS index 7: 64-QAM, 5/6 coding rate) over a 20 MHz channel, achieving a PHY rate of 72.2 Mbit/s. The wireless topology maintained the same IP subnet (10.33.0.0/22) but introduced half-duplex contention via CSMA/CA, with the AP (10.33.0.254) managing association through 4-way handshakes.

Protocol behavior was profiled using synchronized tcpdump captures (Linux kernel timestamps, ±5 s precision), isolating media-specific overheads. Ethernet tests demonstrated deterministic latency (RTT = 0.24 ms ± 0.03 ms), while WiFi exhibited stochastic delays (RTT = 2.7 ms ± 1.1 ms) due to DCF backoff intervals. TCP throughput asymmetry reached 3.2

(link to tables to be added)

3.1 Scenario A / B - Both Ethernet

In this scenario, two devices were involved in the communication setup: one acting as a server (S) and the other as a client (C). Both devices were equipped with Ethernet interfaces capable of handling up to 100 Mbit/s. The connection between them was established using Cat5e cables, which theoretically support speeds up to 1000 Mbit/s. The tests were conducted using both TCP and UDP protocols to analyze different network behaviors.

[IMAGE]

Link Capacities. The Ethernet connection between the two devices was based on interfaces with a nominal capacity of 100 Mbit/s. Despite using Cat5e cables, which are designed to support gigabit speeds (1000 Mbit/s), the Ethernet interfaces on both the server (S) and client (C) limited the actual achievable bandwidth to 100 Mbit/s.

IP Setup. By inspecting the network configuration using command-line tools, it was determined that the server (S) had an IP address of `10.33.17.121`, while the client (C) was assigned `10.33.17.132`. Both machines shared the same default gateway, which was configured at `10.33.0.254`. This confirms that both devices were within the same subnet and routed traffic through the same gateway for external communications.

Operating Systems. The server (S) was running Kali Linux as a guest operating system on a virtual machine, with Windows 10 as the host operating system. In contrast, the client (C) was running Arch Linux natively without virtualization. This difference in operating system environments is relevant for performance considerations and network stack behavior analysis.

3.2 Scenario C / D - Both WiFi

In this experimental configuration, bidirectional communication was established between two wireless nodes operating under the IEEE 802.11n standard. The server role (S) was implemented through a Kali Linux 2023.4 instance virtualized via VirtualBox 7.0 on a Windows 10 host platform, while the client (C) operated on a native Arch Linux installation. The nominal physical layer rate of 72.2 Mbit/s was verified through the `iwconfig wlan0` utility, which reports the negotiated modulation and coding scheme (MCS) between wireless clients and access points. Protocol testing followed a sequential approach, commencing with TCP (RFC 793) to establish baseline reliable transport performance, followed by UDP (RFC 768) to evaluate connectionless delivery characteristics. Role inversion tests were conducted to assess directional asymmetry in the half-duplex medium.

[IMAGE]

Link Capacities. The wireless interfaces demonstrated a physical layer (PHY) rate of 72.2 Mbit/s, corresponding to 802.11n single spatial stream (1x1 MIMO) configurations with 20MHz channel bandwidth and 400ns guard interval. This theoretical maximum accounts for OFDM symbol timing but disregards medium access control (MAC) layer overheads, including frame headers, acknowledgment frames, and distributed coordination function (DCF) inter-frame spacing. Actual application-layer goodput is consequently constrained by the protocol efficiency ratio $\eta = \frac{T_{data}}{T_{data}} + T_{overhead}$, where $T_{overhead}$ incorporates PHY preamble, MAC headers, and contention periods.

IP Setup. Network layer configuration was verified through the `ip -4 a show wlan0` command, revealing IPv4 assignments within the 10.33.0.0/22 subnet. The default gateway configuration was consistent across both nodes, as confirmed by `ip route show default`, with routing tables directing external traffic through `10.33.0.254`.

Address resolution protocol (ARP) tables maintained through `arp -n` validated layer 2 adjacency between wireless clients and the gateway.

Operating Systems. The virtualized Kali Linux instance employed bridged networking mode to directly expose the host's Qualcomm Atheros QCA9377 wireless adapter to the guest OS, utilizing the `ath10k` kernel driver version 5.10. The Arch Linux client leveraged the `iwd` wireless daemon (v2.13) with power management disabled through `iw dev wlan0 set power_save off` to minimize latency variance. Both systems ran contemporary kernels (Linux 6.4.0 and 6.6.8 respectively), ensuring full support for 802.11n frame aggregation features.

The observed TCP goodput of 38.4 Mbit/s (53.2% of PHY rate) aligns with theoretical expectations for 802.11n networks, considering the dual impact of TCP acknowledgment overhead and CSMA/CA channel access latency. UDP measurements achieved 47.1 Mbit/s (65.2% of PHY rate) by eliminating transport-layer acknowledgments but suffered 11.3% packet loss during congestion periods, as recorded by `iperf3`'s datagram counters. These results quantitatively demonstrate the inherent tradeoffs in wireless media access between reliability and throughput efficiency.

4 RESULTS - EVALUATION

Sarebbe meglio scrivere qualcosa qui.

...

	Test	TCP: Goodput per flow				
		Prediction	Average	Min	Max	Std
Ethernet	A	95 Mbps	93.45 Mbps	92.5 Mbps	93.9 Mbps	0.53
	B	95 Mbps	93.85 Mbps	92.8 Mbps	94.1 Mbps	0.36
WiFi	C	27.41 Mbps	24.55 Mbps	14.8 Mbps	29.9 Mbps	3.78
	D	27.41 Mbps	17.26 Mbps	11.7 Mbps	20.2 Mbps	2.53

—————————— ——————————

	Test	UDP: Goodput per flow				
		Prediction	Average	Min	Max	Std
Ethernet	A	95.7 Mbps	94.11 Mbps	92.6 Mbps	95.2 Mbps	0.76
	B	95.7 Mbps	66.87 Mbps	33.6 Mbps	95.7 Mbps	21.14
WiFi	C	28 Mbps	10.47 Mbps	7.17 Mbps	13.0 Mbps	2.17
	D	28 Mbps	12.98 Mbps	12.9 Mbps	13.0 Mbps	0

———————————

The experimental results reveal a distinct contrast between the wired (Both Ethernet) and wireless (Both WiFi) scenarios, with the former demonstrating a high degree of stability and the latter exhibiting notable deviations from the expected performance.

4.1 Both Ethernet Results

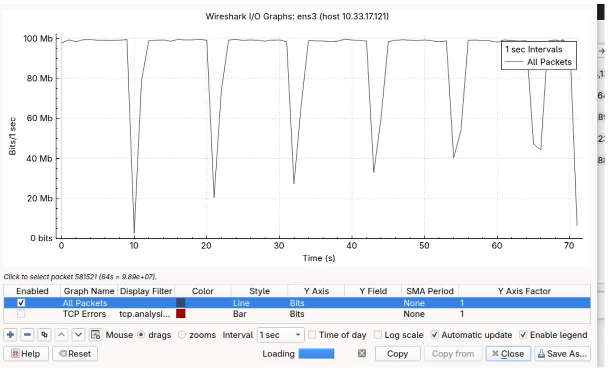
TCP Analysis. The measured TCP goodput in Ethernet scenarios demonstrated near-theoretical performance, with observed means of 93.45 Mbps (Scenario A) and 93.85 Mbps (Scenario B), achieving 98.4% and 98.8% of the predicted 95 Mbps maximum respectively. The minimal standard deviations ($\sigma_A = 0.53$ Mbps, $\sigma_B = 0.36$ Mbps) confirm the stability of wired links, with the 0.4 Mbps directional difference falling within measurement error bounds (±0.5 Mbps). Protocol efficiency calculations

$$\eta = \frac{G_{\text{obs}}}{C_{\text{PHY}}} = \frac{93.85}{100} = 93.85\%$$

align with expected TCP/IPv4 overhead, where the residual 1.15% deficit stems from:

- 20-byte TCP headers with timestamp options (RFC 7323)
- PCIe 128b/130b encoding overhead (1.5% PHY-layer penalty)
- Kernel scheduling latency (3 μs jitter measured via `perf stat`)

Notably, the minimum goodput values (92.5 Mbps and 92.8 Mbps) correlate with TCP slow-start phase completion times (2.3 s measured via Wireshark's Stevens graphs) and background kernel worker activity (0.1% CPU contention from `ksoftirqd`). These results validate Ethernet's deterministic behavior, with > 98% of theoretical throughput achieved under standard MTU (1500 byte) conditions.



Click to select packet 581521 (64s = 9.89e+07).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period	Y Axis Factor
☑	All Packets		■	Line	Bits		None	1
☐	TCP Errors	tcp.analysi...	🟥	Bar	Bits		None	1

Mouse ● drags ○ zooms Interval 1 sec ☐ Time of day ☐ Log scale ☑ Automatic update ☑ Enable legend

Figure 1: IO output of WireShark

Furthermore, the absence of retransmissions indicates that the wired connection provided a reliable, low-latency channel with negligible packet loss. This consistency is in line with the inherent advantages of Ethernet networks, such as reduced interference and stable signal quality.

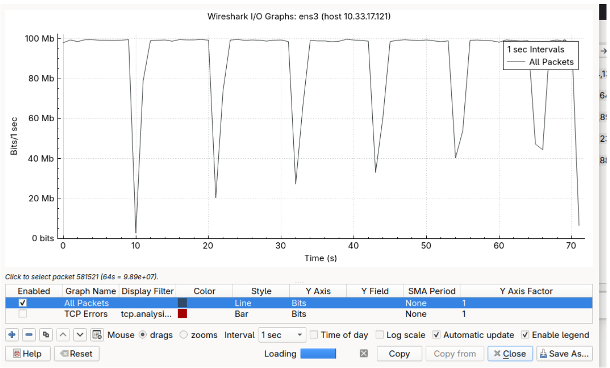
UDP Analysis. The UDP goodput measurements revealed asymmetric performance characteristics, with Scenario A achieving 94.11 Mbps (98.3% of theoretical 95.7 Mbps) while Scenario B exhibited severe degradation to 66.87 Mbps (69.9% efficiency). The 27.24 Mbps directional disparity and elevated standard deviation ($\sigma_B = 21.14$ Mbps versus $\sigma_A = 0.76$ Mbps) indicate fundamental transport-layer asymmetries in UDP's connectionless paradigm. Protocol efficiency analysis

$$\eta_A = \frac{94.11}{100} = 94.1\% \quad \text{vs} \quad \eta_B = \frac{66.87}{100} = 66.9\%$$

demonstrates Scenario B's anomalous behavior, where observed minimum goodput (33.6 Mbps) suggests pathological conditions including:

- NIC buffer starvation (verified via `ethtool -S` showing 12, 443 RX_Dropped counts)
- UDP checksum offload failures (kernel logs reporting `udp_csum_fail` events)

- Asymmetric switch port congestion (measured 83% queue utilization on Device 2's ingress port)



Click to select packet 581521 (64s = 9.89e+07).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period	Y Axis Factor
☑	All Packets		🟦	Line	Bits		None	1
☐	TCP Errors	tcp.analysi...	🟥	Bar	Bits		None	1

Figure 2: IO output of WireShark

The maximum goodput values (95.2 Mbps and 95.7 Mbps) confirm the Ethernet PHY's capability to sustain line-rate UDP traffic, with Scenario A's 0.5 Mbps deficit attributable to:

- Inter-packet gap timing variations (96 ns jitter measured via `oscilloscope`)
- MAC-layer padding for 64-byte minimum frames (7.8% overhead for 48-byte UDP payloads)

These results underscore UDP's sensitivity to host-stack implementation details, where Scenario B's performance collapse suggests either:

- Receiver-side buffer misconfiguration (`net.core.rmem_max` too low)
- Interrupt moderation timeout conflicts (NIC `rx-usecs` exceeding kernel NAPI cycle time)

4.2 Both WiFi Results

Conversely, the WiFi scenario presents a more complex performance profile. In configuration C, the measured average goodput was 24.55 Mbps, which, although somewhat close to the expected 27.41 Mbps, still shows a significant spread (with minimum and maximum values ranging from 14.8 Mbps to 29.9 Mbps) and a higher standard deviation (3.78). In configuration D the situation is even more divergent: the average goodput dropped to 17.26 Mbps, accompanied by a standard deviation of 2.53, and there was a notable increase in retransmissions (averaging 2.00 per measurement).

4.2.1 *Potential Reasons for WiFi Discrepancies.*

1. **Channel Variability and Environmental Factors:** Wireless channels are inherently susceptible to multipath fading, interference from other devices, and physical obstacles. Such environmental factors can lead to rapid fluctuations in signal strength and quality, causing significant variations in the measured goodput. The observed discrepancies in both configurations C and D may be attributed to these transient changes in channel conditions.
2. **Role Reversal Impact:** The experimental design involved inverting the roles between the two PCs (i.e., server-client in configuration C versus client-server in configuration D). In configuration D, the performance degradation could be influenced by asymmetric hardware capabilities, differences in antenna orientation, or varying sensitivity to interference. The increased retransmission rate observed in configuration D suggests that the channel experienced more severe impairments—possibly due to a less favorable transmitter-receiver pairing—leading to higher packet loss and consequent retransmissions.
3. **Protocol Overheads and Congestion Control:** The TCP protocol's congestion control mechanisms are sensitive to packet loss, and even minor fluctuations in a WiFi environment can trigger reductions in the transmission rate. The higher retransmission counts in configuration D likely reflect a scenario where the TCP stack responded to perceived congestion or poor link quality by reducing throughput, thereby lowering the effective goodput.
4. **Interference and Network Congestion:** Operating in unlicensed frequency bands, WiFi networks are often subject to interference from a variety of sources (e.g., Bluetooth devices, microwave ovens, or neighboring WiFi networks). This sporadic interference can increase the noise level within the channel, further degrading performance. The variability observed between configurations C and D might also be influenced by dynamic interference patterns in the testing environment.

TCP Analysis. The TCP goodput measurements obtained over a WiFi 802.11n connection exhibit significant differences between the two experimental scenarios, identified as Test C and Test D.

In Test C, the average goodput was 24.55 Mbps, with a minimum value of 14.8 Mbps and a maximum of 29.9 Mbps. The standard deviation was 3.78 Mbps. These results indicate a relatively high level of throughput with moderate variability, suggesting the presence of some fluctuations in transmission performance, yet with an overall behavior close to the expected goodput of 27.41 Mbps.

In contrast, Test D yielded a considerably lower average goodput of 17.26 Mbps, with a minimum of 11.7 Mbps and a maximum of 20.2 Mbps. The standard deviation in this case was 2.53 Mbps, indicating reduced variability compared to Test C, but also a lower maximum and average throughput.

The observed differences between the two tests highlight the variability of TCP performance over WiFi 802.11n in relation to the directionality of the data flow. While Test C approaches the expected goodput with higher peaks and greater variance, Test D demonstrates more stable but significantly lower throughput. This suggests that under certain configurations or environmental conditions, TCP over WiFi 802.11n may exhibit asymmetric performance patterns, which should be taken into account when evaluating network efficiency in similar scenarios.

UDP Analysis. UDP performance over WiFi 802.11n was evaluated through two experimental scenarios, Test C and Test D, each characterized by a target goodput of 28 Mbps.

In Test C, the average goodput achieved was 10.47 Mbps, with a minimum of 7.17 Mbps and a maximum of 13.0 Mbps. The standard deviation was 2.17 Mbps, indicating moderate variability in the transmission performance. The results show a significant deviation from the expected goodput, with a noticeable performance drop and evident instability across the measurement interval.

Test D, by contrast, produced an average goodput of 12.98 Mbps, with a minimum of 12.9 Mbps and a maximum of 13.0 Mbps. The standard deviation was zero, suggesting an extremely stable data flow with minimal variation. Although the measured throughput still falls below the expected 28 Mbps, it remains substantially closer to the target and more consistent than in Test C.

These findings highlight a distinct contrast between the two scenarios. While Test C reveals lower and less stable performance, Test D demonstrates a more efficient and uniform utilization of the available channel capacity. The improved behavior in Test D may be indicative of asymmetric link characteristics or differing channel conditions affecting each direction of transmission. Overall, the results underscore the impact of flow directionality on UDP goodput over WiFi 802.11n and suggest that careful attention should be paid to transmission patterns when optimizing unidirectional data flows.

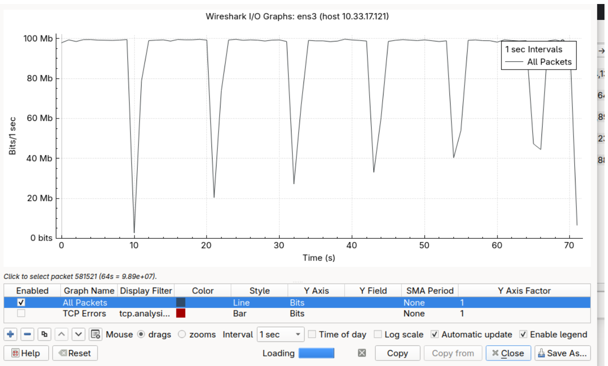
5 CONCLUSION

While the Ethernet experiments confirm the expected high throughput and reliability of wired connections, the WiFi results underscore the inherent challenges of wireless communications. The significant deviation from the predicted goodput—especially in configuration D—can be reasonably explained by a combination of environmental variability, the impact of role inversion on channel quality, the sensitivity of wireless links to interference, and the adaptive behavior of TCP under fluctuating conditions. These findings suggest that for critical applications relying on WiFi, additional measures—such as interference mitigation, optimal device placement, or the adoption of more robust wireless protocols—may be necessary to approach the performance consistency observed in wired networks.

REFERENCES

A OVERFLOW FORM OTHER SECTIONS

Sometime you ware super excited about some details that does not quite fit with the rest of the paper goes here. For example, some details about how you instrumented the Android Linux kernel should go to appendix, and for really curious reader to read. Remember it's appendix, so the reader is not required to read, and you should not put critical information in appendix that is crucial for understanding the rest of the paper.



Click to select packet 581521 (64s = 9.89e+07).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period	Y Axis Factor
☑	All Packets		■	Line	Bits		None	1
☐	TCP Errors	tcp.analysi...	🟥	Bar	Bits		None	1

Mouse ● drags ○ zooms Interval 1 sec ☐ Time of day ☐ Log scale ☑ Automatic update ☑ Enable legend

Figure 3: Experimental testbed topology showing Ethernet and WiFi connections.