

# Showcase - 802.11 Wireless Handover

**Avaliação de Desempenho de Sistemas - 29 de Julho de 2025**

Arthur Cadore M. Barcella , Deivid Fortunato Frederico

# Sumário

Parte 2 - Projeto de ADS incluindo trafegos TDP e UDP ao showcase .....	3
A Simulação .....	9
Análise com Iperf .....	31
Conclusão .....	46

## **Parte 2 - Projeto de ADS incluindo trafegos TDP e UDP ao showcase**

## Objetivo do Experimento

- Avaliar o impacto do handover Wi-Fi em comunicações TCP e UDP
- Simular perdas de pacotes
- Simular variações de vazão
- Simular comportamento sob mobilidade.

<https://inet.omnetpp.org/docs/showcases/wireless/handover/doc/index.html>

# Fatores Variáveis

- Tipo de protocolo: TCP vs UDP
- Velocidade do host móvel
- Distância entre os APs
- Potência de transmissão

# Métricas de Avaliação:

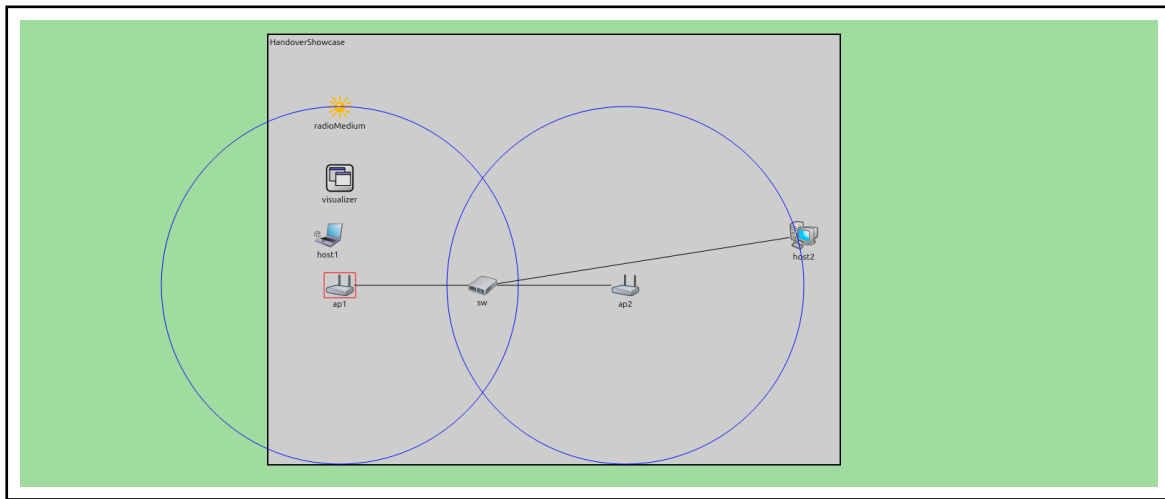
- Taxa de perda de pacotes UDP
- Vazão TCP (throughput)
- Tempo de resposta / latência
- Número de retransmissões TCP (implícito via logs)
- Momentos e duração do handover

# Modelos de Simulação usados

- Plataforma: INET Framework para OMNeT++
- Modelo de mobilidade: LinearMobility
- Tecnologia de acesso: IEEE 802.11 (Wi-Fi) com activeScan e dois APs em canais distintos
- Aplicações:
  - Aplicação UDP: UDPBasicApp e UDPSink
  - Aplicação TCP: TcpBasicClient e TcpBasicServer
- Modelo físico: potência de transmissão de 2 mW, interferência parcial desabilitada

# Topologia

Figura 1: Elaborada pelo Autor



Cenário de Handover



# A Simulação

# Configuração do Cenário

```
1 network HandoverShowcase
2 {
3     parameters:
4         @display("bgb=800,600");
5     types:
6         channel Eth100M extends ned.DatarateChannel {
7             datarate = 100Mbps;
8             delay = 0.1ms;
9         }
10
11     [...]
12
13     host2: StandardHost {
```

## Configuração do Cenário (ii)

```
14         parameters:
15             @display("p=750,280;r=,,#707070");
16     }
17
18     [...]
19
20     sw: EthernetSwitch {
21         parameters:
22             @display("p=300,350;r=,,#707070");
23     }
24
25     connections allowunconnected:
26         ap1.ethg++ <--> Eth100M <--> sw.ethg++;
```

## Configuração do Cenário (iii)

```
27         ap2.ethg++ <--> Eth100M <--> sw.ethg++;  
28         host2.ethg++ <--> Eth100M <--> sw.ethg++;  
29     }
```

# Configurações adicionais de Simulação

```
1 # Ativa bridging L2 nos APs entre wlan e eth
2 **.ap1.hasEthernetBridging = true
3 **.ap2.hasEthernetBridging = true
4
5 # IPs estáticos e rotas padrão
6 *.host1.ipv4.address = "10.0.0.2"
7 *.host1.ipv4.netmask = "255.255.255.0"
8 *.host1.wlan[0].mgmt.associateToSsid = "AP1"
9
10 *.host2.ipv4.address = "10.0.0.3"
11 *.host2.ipv4.netmask = "255.255.255.0"
12 *.host2.ipv4.defaultGateways = "10.0.0.1"
13
```

## Configurações adicionais de Simulação (ii)

```
14 # MAC address manual
15 *.host1.eth[0].macAddress = "00:00:00:00:00:01"
16 *.ap1.eth[0].macAddress = "00:00:00:00:00:02"
17 *.ap2.eth[0].macAddress = "00:00:00:00:00:03"
18 *.host2.eth[0].macAddress = "00:00:00:00:00:04"
19 *.switch.eth[*].macAddress = auto
20
21 # Ativa pingApp em host2 para testar conectividade
22 *.host2.numApps = 1
23 *.host2.app[0].typename = "PingApp"
24 *.host2.app[0].destAddr = "10.0.0.1" # ping do host2 para
host1
25 *.host2.app[0].startTime = 1s
```

## Configurações adicionais de Simulação (iii)

```
26 *.host2.app[0].count = 5
```

# Configurações de TCP e UDP

```
1 # Aplicações UDP no host1 (cliente) e host2 (sink)
2 *.host1.numUdpApps = 1
3 *.host1.udpApp[0].typename = "UDPBasicApp"
4 *.host1.udpApp[0].destAddresses = "host2"
5 *.host1.udpApp[0].destPort = 1000
6 *.host1.udpApp[0].messageLength = exponential(1000B)
7 *.host1.udpApp[0].sendInterval = 0.1s
8 *.host1.udpApp[0].startTime = 1s
9 *.host1.udpApp[0].stopTime = 99s
10 *.host1.udpApp[0].verbose = true
11 *.host1.udpApp[0].packetName = "UDPData"
12 *.host1.udpApp[0].numPk = 1000
13
```



## Configurações de TCP e UDP (ii)

```
14 *.host2.numUdpApps = 1
15 *.host2.udpApp[0].typename = "UDPSink"
16 *.host2.udpApp[0].localPort = 1000
17 *.host2.udpApp[0].verbose = true
18
19
20 # Aplicações TCP no host1 (cliente) e host2 (servidor)
21 *.host1.numTcpApps = 1
22 *.host1.tcpApp[0].typename = "TcpBasicClient"
23 *.host1.tcpApp[0].connectAddress = "host2"
24 *.host1.tcpApp[0].connectPort = 2000
25 *.host1.tcpApp[0].startTime = 1s
26 *.host1.tcpApp[0].stopTime = 99s
```

## Configurações de TCP e UDP (iii)

```
27 *.host1.tcpApp[0].verbose = true
28 *.host1.tcpApp[0].messageLength = 1000B
29 *.host1.tcpApp[0].sendBytes = 1000000
30
31 *.host2.numTcpApps = 1
32 *.host2.tcpApp[0].typename = "TcpBasicServer"
33 *.host2.tcpApp[0].localPort = 2000
34 *.host2.tcpApp[0].verbose = true
```

## Verbose pra export em logs

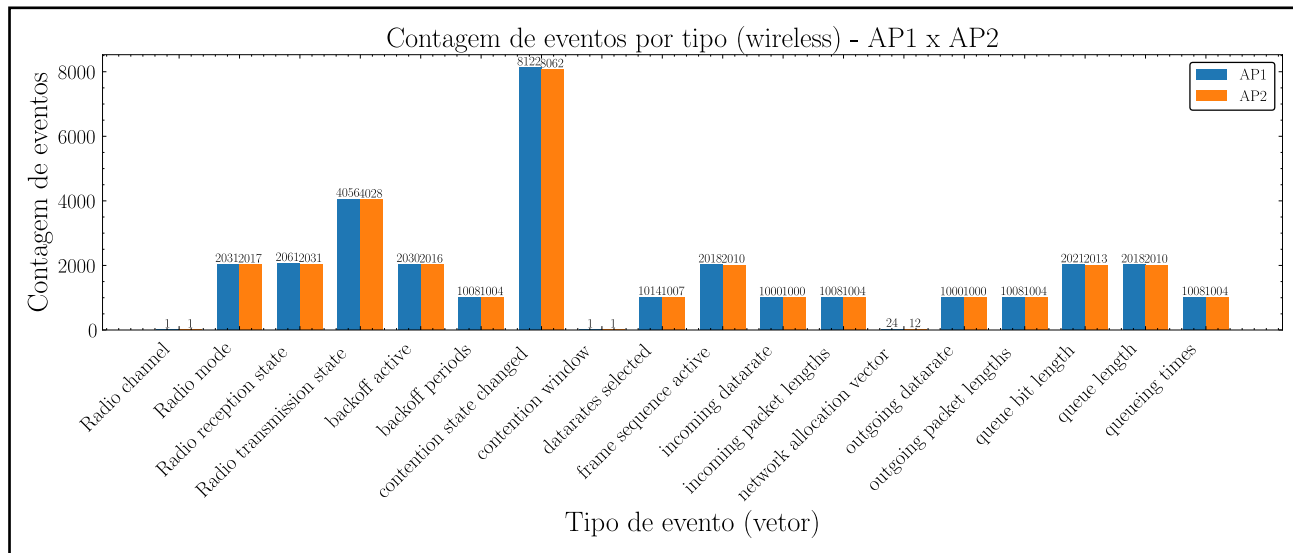
```
1 # Estatísticas UDP e TCP para hosts
2 *.host1.udpApp[0].recordPacketsSent = true
3 *.host1.udpApp[0].recordBytesSent = true
4 *.host1.udpApp[0].recordPacketsReceived = true
5 *.host1.udpApp[0].recordBytesReceived = true
6 *.host2.udpApp[0].recordPacketsReceived = true
7 *.host2.udpApp[0].recordBytesReceived = true
8
9 *.host1.tcpApp[0].recordPacketsSent = true
10 *.host1.tcpApp[0].recordBytesSent = true
11 *.host1.tcpApp[0].recordPacketsReceived = true
12 *.host1.tcpApp[0].recordBytesReceived = true
13 *.host2.tcpApp[0].recordPacketsReceived = true
```

## Verbose pra export em logs (ii)

```
14 *.host2.tcpApp[0].recordBytesReceived = true
15
16
17 # Estatísticas em vetor para aplicações e camadas Wi-Fi
18 *.host1.udpApp[*].statistic-recording = "vector"
19 *.host2.udpApp[*].statistic-recording = "vector"
20 *.host1.tcpApp[*].statistic-recording = "vector"
21 *.host2.tcpApp[*].statistic-recording = "vector"
22 *.host1.wlan[*].agent.statistic-recording = "vector"
23 *.host1.wlan[*].radio.statistic-recording = "vector"
24 *.host1.wlan[*].mac.statistic-recording = "vector"
```

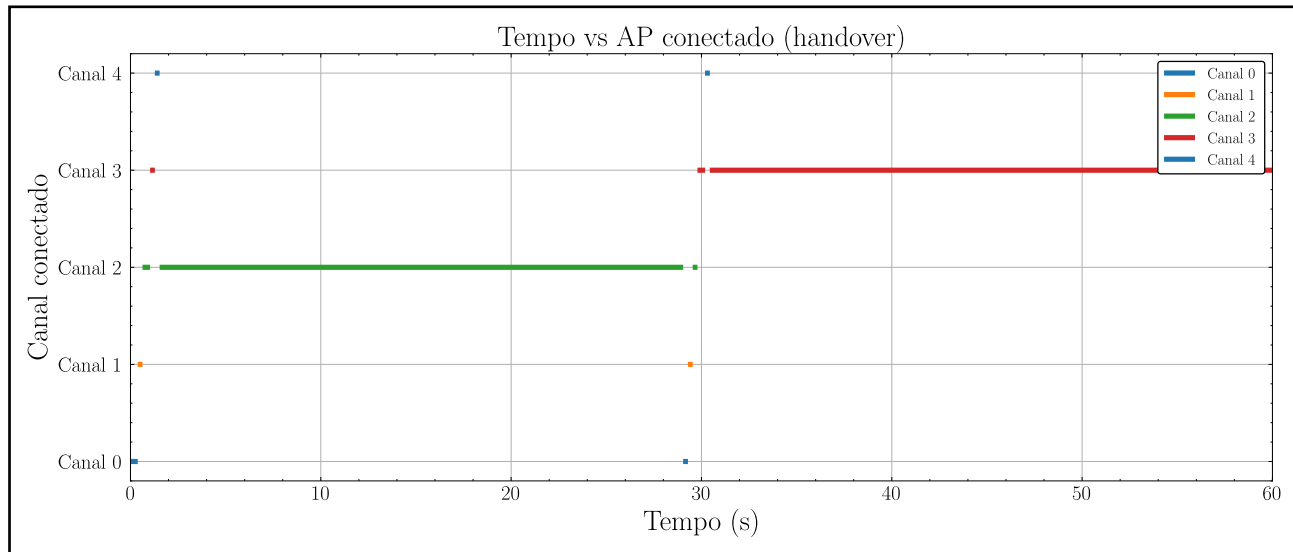
# Contagem de eventos

Figura 2: Elaborada pelo Autor



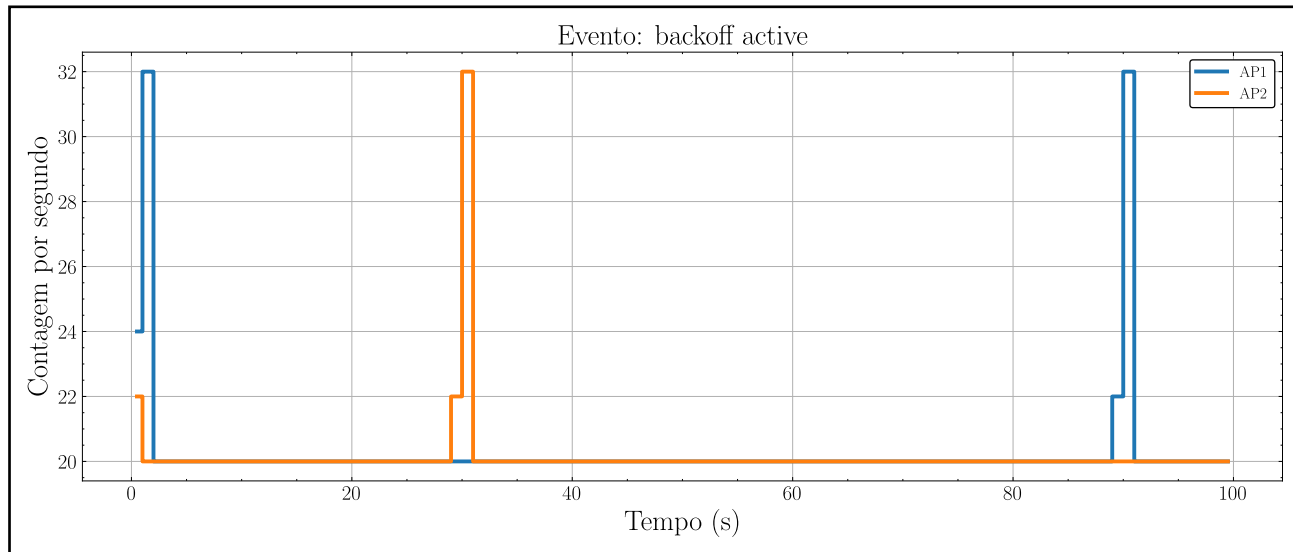
# Handover

Figura 3: Elaborada pelo Autor



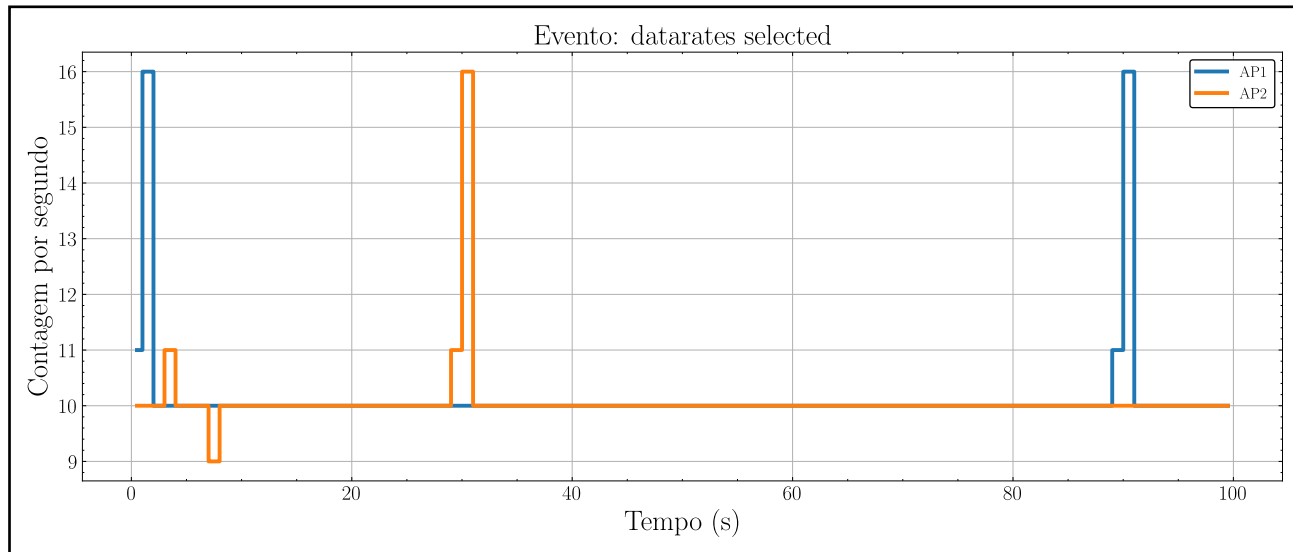
# Eventos de Backoff

Figura 4: Elaborada pelo Autor



# Taxa de Dados e Frame Sequence

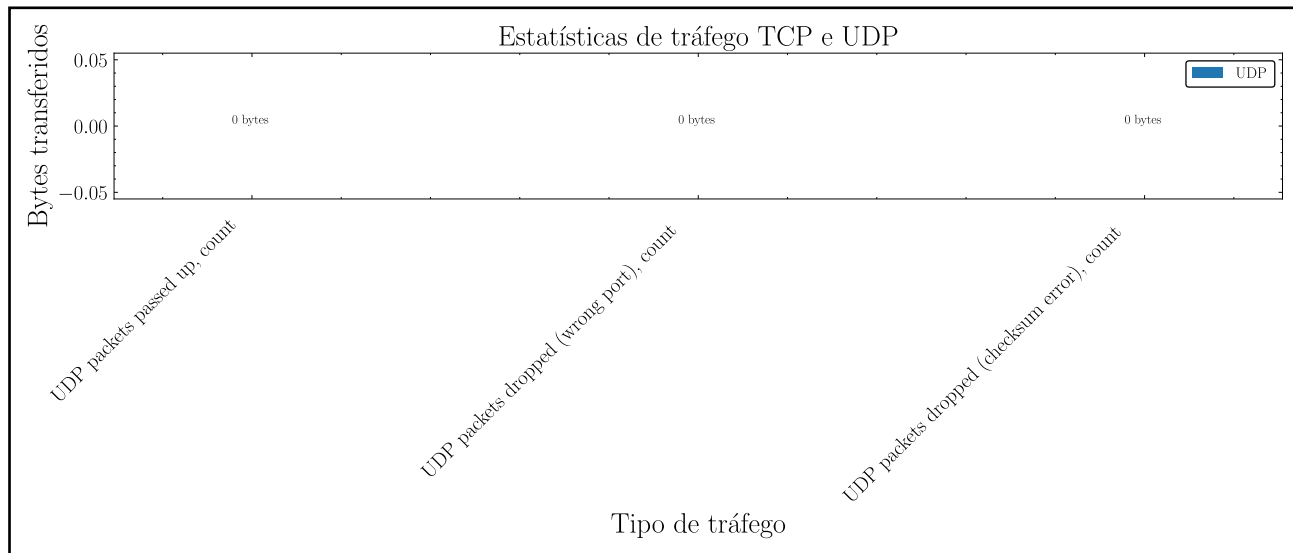
Figura 5: Elaborada pelo Autor





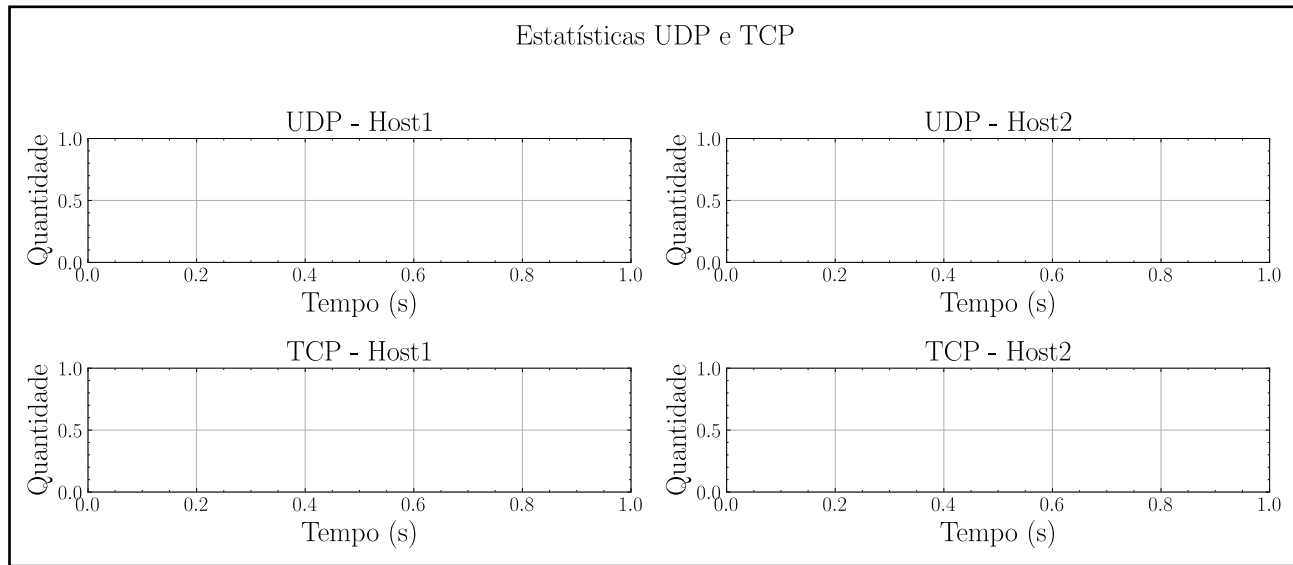
# Estatísticas de Tráfego

Figura 6: Elaborada pelo Autor



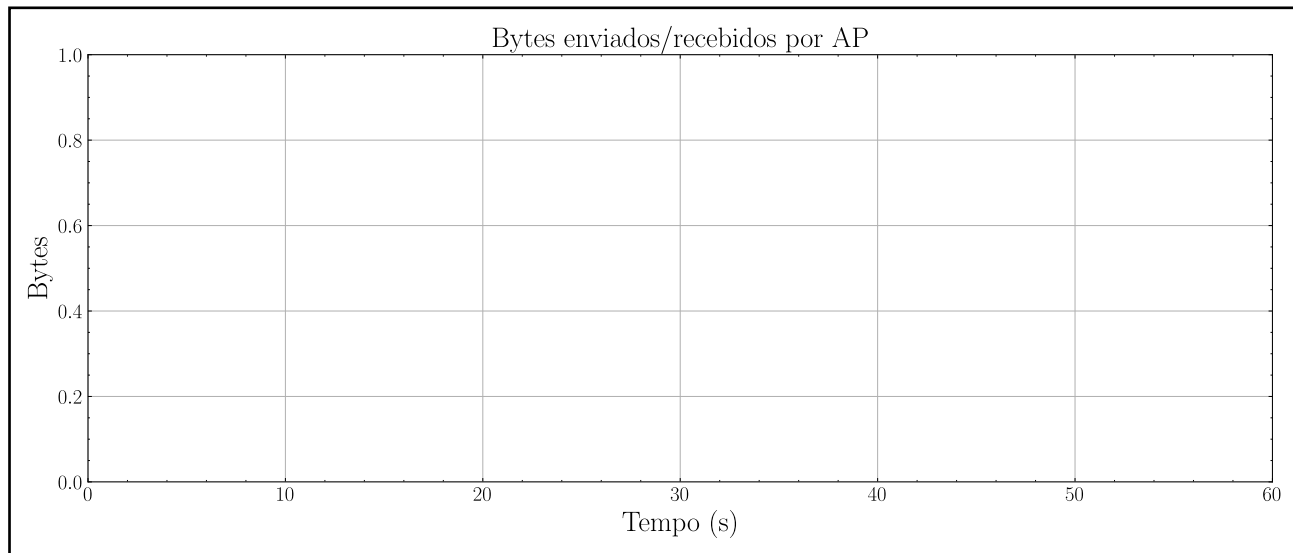
# Estatísticas UDP/TCP

Figura 7: Elaborada pelo Autor



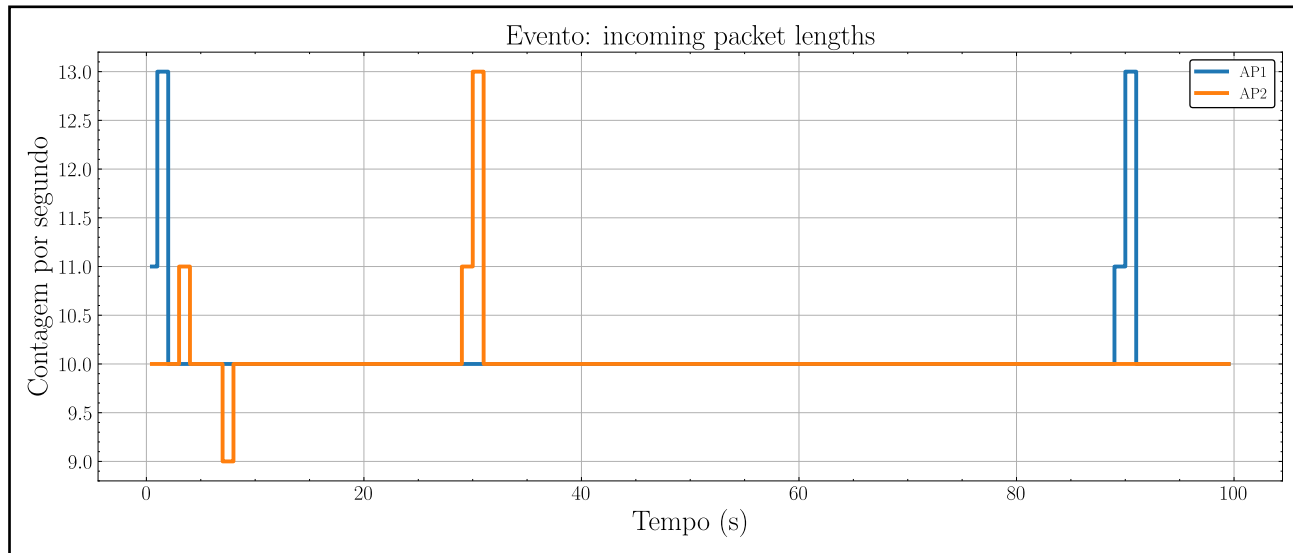
# Estatísticas de Bytes

Figura 8: Elaborada pelo Autor



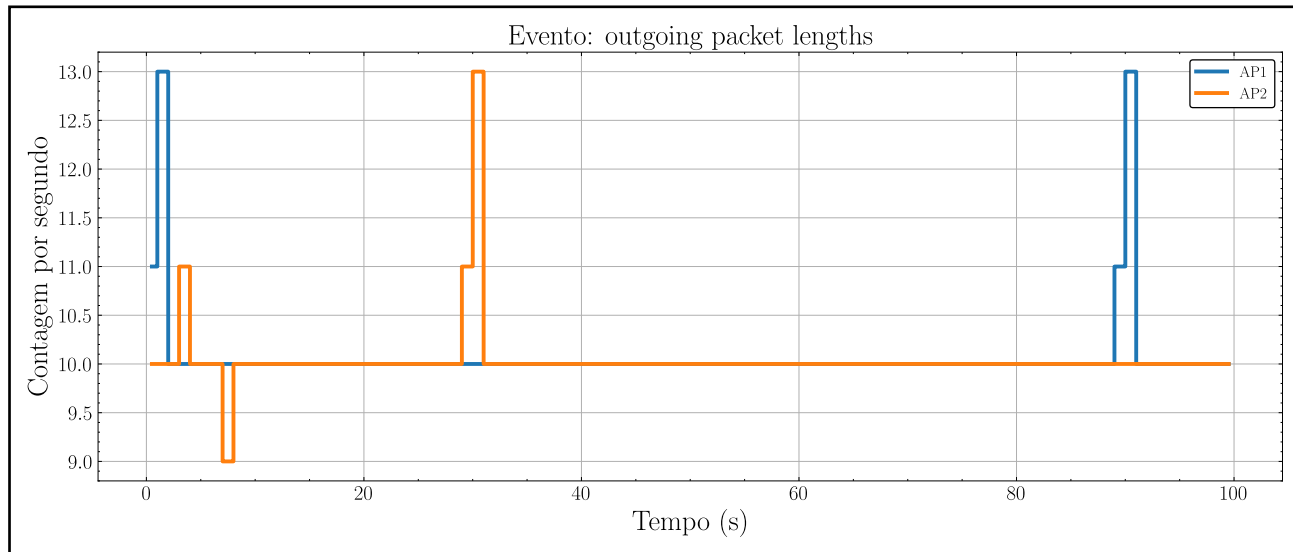
# Taxa de Dados In/out

Figura 9: Elaborada pelo Autor



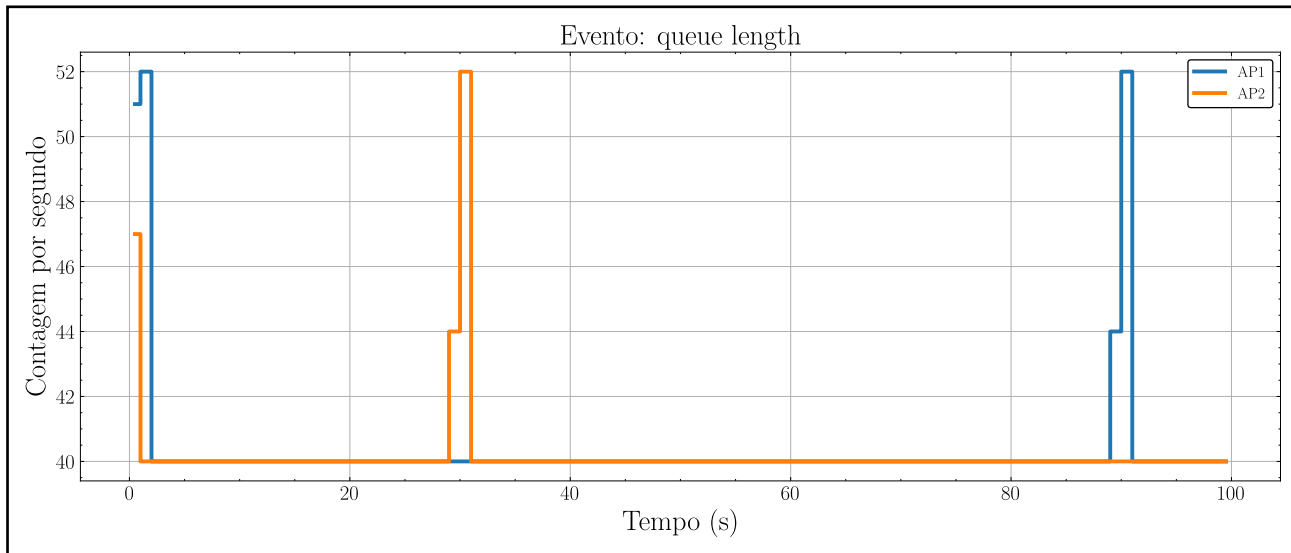
# Taxa de Dados In/out

Figura 10: Elaborada pelo Autor



# Análise de Filas

Figura 11: Elaborada pelo Autor



# **Análise com Iperf**

# Cenário

Figura 12: Elaborada pelo Autor



Cenário de Teste com Iperf



# Configuração de interface e VLAN

```
1 [SW-CADORE]vlan 1234
2 [SW-CADORE-vlan1234]exit
3
4
5 [SW-CADORE]interface range GigabitEthernet 1/0/9 to
GigabitEthernet 1/0/11
6 [SW-CADORE-if-range]port link-type access
7 [SW-CADORE-if-range]port access vlan 1234
```

# Verificação de interface

```
1 [SW-CADORE]display interface GigabitEthernet 1/0/9 brief
2 Brief information on interfaces in bridge mode:
3 Link: ADM - administratively down; Stby - standby
4 Speed: (a) - auto
5 Duplex: (a)/A - auto; H - half; F - full
6 Type: A - access; T - trunk; H - hybrid
7 Interface          Link Speed    Duplex Type PVID
8 Description
GE1/0/9             UP    1G(a)    F(a)   A    1234
```

# Tabela ARP

```
1 [SW-CADORE-Vlan-interface1234]ping 172.16.10.2
2 Ping 172.16.10.2 (172.16.10.2): 56 data bytes, press CTRL+C
3 to break
4 56 bytes from 172.16.10.2: icmp_seq=0 ttl=64 time=3.562 ms
5 [SW-CADORE-Vlan-interface1234]display arp vlan 1234
6   Type: S-Static    D-Dynamic    0-Openflow    R-Rule    M-
7   Multiport    I-Invalid
8   IP address      MAC address      VLAN/VSI name Interface
   Aging Type
172.16.10.1      8c47-be17-8271 1234          GE1/0/9
1191    D
```

## Tabela ARP (ii)

9	172.16.10.2	2800-afb5-b278 1234	GE1/0/11
	1189 D		

# Teste de Conectividade

Figura 13: Elaborada pelo Autor

No.	Time	Source	Destination	Protocol	Length	Info
62	27.85745...	Dell_17:82:71	Broadcast	ARP	60	Who has 172.16.10.2? Tell 172.16.10.1
63	27.85748...	Dell_b5:b2:78	Dell_17:82:71	ARP	42	172.16.10.2 is at 28:00:af:b5:b2:78
64	27.85835...	172.16.10.1	172.16.10.2	ICMP	74	Echo (ping) request id=0x0001, seq=1111/22276, ttl=128 (reply in 65)
65	27.85841...	172.16.10.2	172.16.10.1	ICMP	74	Echo (ping) reply id=0x0001, seq=1111/22276, ttl=64 (request in 64)
68	28.87418...	172.16.10.1	172.16.10.2	ICMP	74	Echo (ping) request id=0x0001, seq=1112/22532, ttl=128 (reply in 69)
69	28.87425...	172.16.10.2	172.16.10.1	ICMP	74	Echo (ping) reply id=0x0001, seq=1112/22532, ttl=64 (request in 68)
71	29.89087...	172.16.10.1	172.16.10.2	ICMP	74	Echo (ping) request id=0x0001, seq=1113/22788, ttl=128 (reply in 72)
72	29.89093...	172.16.10.2	172.16.10.1	ICMP	74	Echo (ping) reply id=0x0001, seq=1113/22788, ttl=64 (request in 71)
75	30.90812...	172.16.10.1	172.16.10.2	ICMP	74	Echo (ping) request id=0x0001, seq=1114/23044, ttl=128 (reply in 76)
76	30.90819...	172.16.10.2	172.16.10.1	ICMP	74	Echo (ping) reply id=0x0001, seq=1114/23044, ttl=64 (request in 75)

Frame 64: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0	0000	28 00 af b5 b2 78 8c 47 be 17 82 71 08 00 45 00
Ethernet II, Src: Dell_17:82:71 (8c:47:be:17:82:71), Dst: Dell_b5:b2:78	0010	00 3c cc 73 00 00 80 01 02 2a ac 10 0a 01 ac 10
Internet Protocol Version 4, Src: 172.16.10.1, Dst: 172.16.10.2	0020	0a 02 08 00 49 04 00 01 04 57 61 62 63 64 65 66
Internet Control Message Protocol	0030	67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76
	0040	77 61 62 63 64 65 66 67 68 69

Teste de Conectividade

# Configuração de Speed:

```
1 [SW-CADORE]interface range GigabitEthernet 1/0/9 to
2 GigabitEthernet 1/0/11
3 [SW-CADORE-if-range]sp
4 [SW-CADORE-if-range]speed ?
5     10      Specify speed as 10 Mbps
6     100     Specify speed as 100 Mbps
7     1000    Specify speed as 1000 Mbps
8     auto    Enable port's speed negotiation automatically
9 [SW-CADORE-if-range]speed 10
```

# Teste de Iperf:

```
1 SERVER_IP="172.16.10.1" # IP do servidor
2 TEMPO=1000
3
4 echo "Iniciando teste UDP (9 Mbps)..."
5 iperf3 -c $SERVER_IP -u -b 9M -t $TEMPO > udp_result.txt &
6
7 sleep 1 # Pequeno atraso para evitar conflito de conexão
8
9 echo "Iniciando teste TCP..."
10 iperf3 -c $SERVER_IP -t $TEMPO > tcp_result.txt
11
12 echo "Testes finalizados. Resultados em udp_result.txt e
   tcp_result.txt"
```

# Flows

Figura 14: Elaborada pelo Autor

Time	NewH3Techno_d8:b1:10	Spanning-tree-(for-bridges)_00	172.16.10.2	172.16.10.1	LLDP_Multicast	Comment
6.513138557			47078	5201		TCP: 47078 → 5201 [ACK] Seq=38 Ack=710001 Wi...
6.513954168			49837	5202		UDP: 5202 → 49837 Len=1000
6.514820587			47078	5201		TCP: 5201 → 47078 [PSH, ACK] Seq=710001 Ack=...
6.514865812			47078	5201		TCP: 47078 → 5201 [ACK] Seq=38 Ack=710001 Wi...
6.515673523			49837	5202		UDP: 5202 → 49837 Len=1000
6.516522451			49837	5202		UDP: 5202 → 49837 Len=1000
6.517418248			47078	5201		TCP: 5201 → 47078 [PSH, ACK] Seq=711001 Ack=...
6.517452221			47078	5201		TCP: 47078 → 5201 [ACK] Seq=38 Ack=712001 Wi...
6.518249017			47078	5201		TCP: 5201 → 47078 [PSH, ACK] Seq=712001 Ack=...
6.518268122			47078	5201		TCP: 47078 → 5201 [ACK] Seq=38 Ack=713001 Wi...
6.519112786			47078	5201		TCP: 5201 → 47078 [PSH, ACK] Seq=713001 Ack=...
6.519157049			47078	5201		TCP: 47078 → 5201 [ACK] Seq=38 Ack=714001 Wi...
6.519965273			49837	5202		UDP: 5202 → 49837 Len=1000
6.520819302			49837	5202		UDP: 5202 → 49837 Len=1000
6.521670064			49837	5202		UDP: 5202 → 49837 Len=1000
6.522530721			47078	5201		TCP: 5201 → 47078 [PSH, ACK] Seq=714001 Ack=...
6.522559472			47078	5201		TCP: 47078 → 5201 [ACK] Seq=38 Ack=715001 Wi...
6.523399647			47078	5201		TCP: 5201 → 47078 [PSH, ACK] Seq=715001 Ack=...
6.523435191			47078	5201		TCP: 47078 → 5201 [ACK] Seq=38 Ack=716001 Wi...
6.524257416			47078	5201		TCP: 5201 → 47078 [PSH, ACK] Seq=716001 Ack=...
6.524301437			47078	5201		TCP: 47078 → 5201 [ACK] Seq=38 Ack=717001 Wi...
6.525110480			49837	5202		UDP: 5202 → 49837 Len=1000
6.525961433			49837	5202		UDP: 5202 → 49837 Len=1000
6.526813832			49837	5202		UDP: 5202 → 49837 Len=1000

Flows TCP e UDP

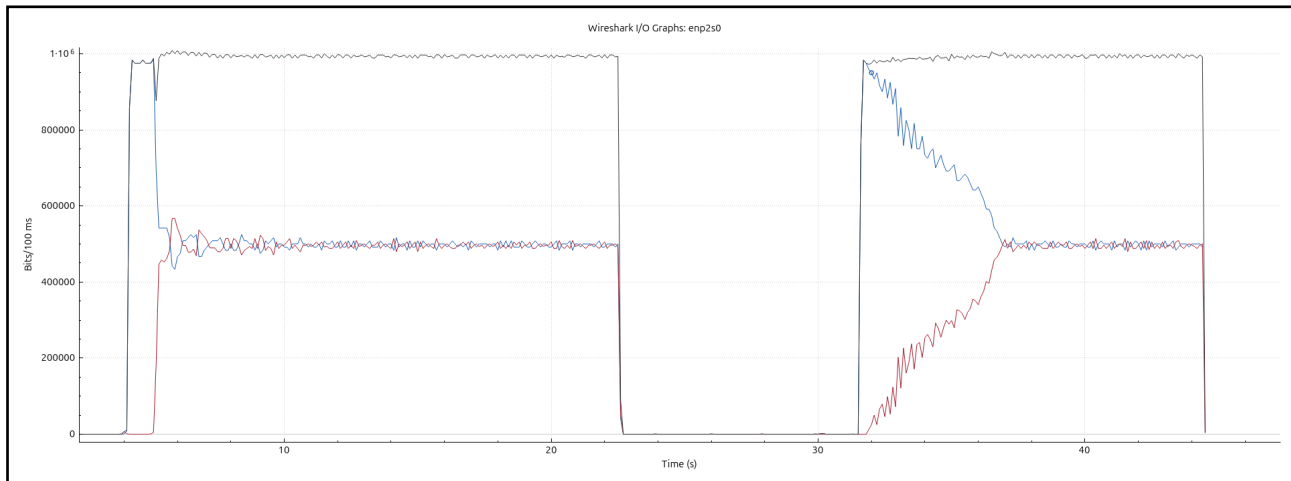


# Simular queda do link

```
1 [SW-CADORE]interface GigabitEthernet 1/0/11
2 [SW-CADORE-GigabitEthernet1/0/11]port access vlan 10
3 [SW-CADORE-GigabitEthernet1/0/11]port access vlan 1234
```

# Captura Wireshark

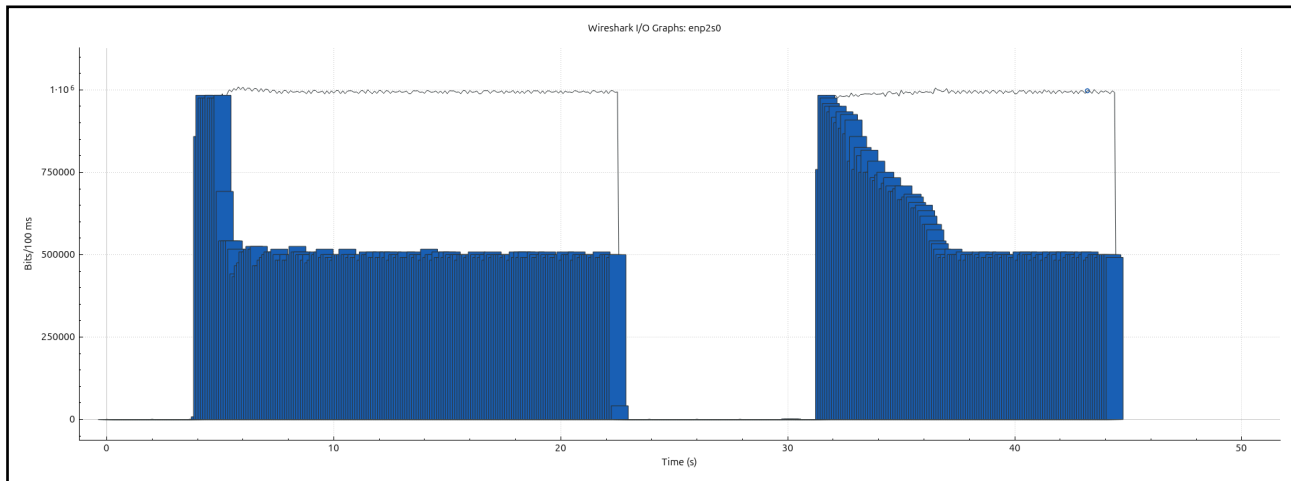
Figura 15: Elaborada pelo Autor



Captura Wireshark

# Trafego UDP

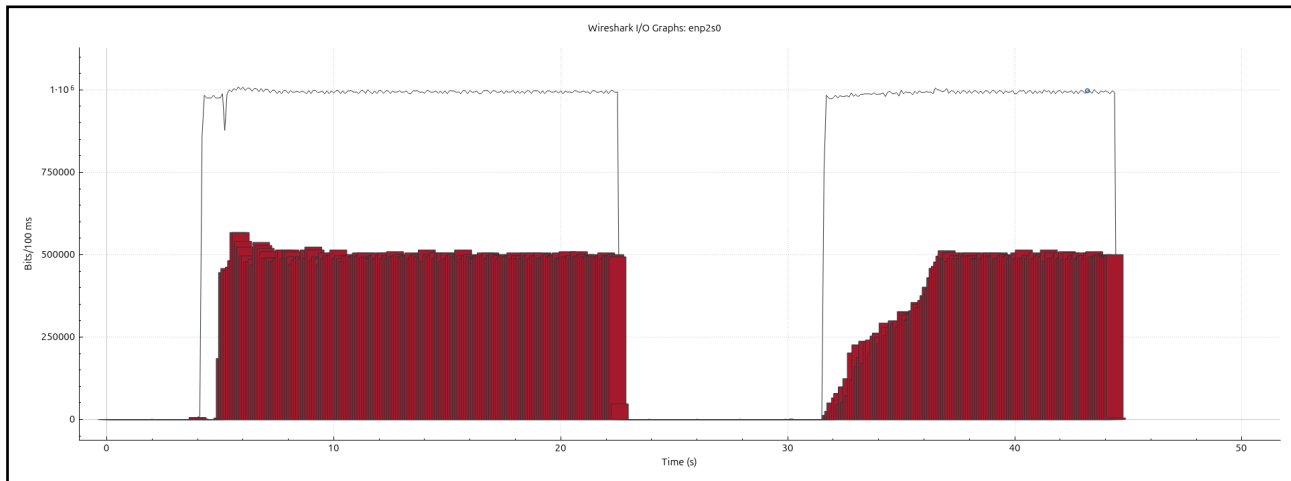
Figura 16: Elaborada pelo Autor



Captura Wireshark do Iperf UDP

# Captura TCP

Figura 17: Elaborada pelo Autor



Captura Wireshark TCP

# Resultados no iperf3

Figura 18: Elaborada pelo Autor

[ 5]	84.00-85.01	sec	580	KBytes	4.74	Mbits/sec	3.842	ms	626/1220	(51%)
[ 5]	85.01-86.00	sec	580	KBytes	4.78	Mbits/sec	4.807	ms	673/1267	(53%)
[ 5]	86.00-87.00	sec	583	KBytes	4.78	Mbits/sec	2.648	ms	724/1321	(55%)
[ 5]	87.00-88.00	sec	584	KBytes	4.78	Mbits/sec	2.656	ms	774/1372	(56%)
[ 5]	88.00-89.00	sec	576	KBytes	4.70	Mbits/sec	2.716	ms	829/1419	(58%)
[ 5]	89.00-90.00	sec	575	KBytes	4.73	Mbits/sec	3.586	ms	873/1462	(60%)
[ 5]	90.00-91.00	sec	585	KBytes	4.77	Mbits/sec	0.791	ms	930/1529	(61%)
[ 5]	91.00-92.00	sec	581	KBytes	4.77	Mbits/sec	2.813	ms	245/840	(29%)
[ 5]	92.00-93.00	sec	579	KBytes	4.75	Mbits/sec	2.573	ms	167/760	(22%)
[ 5]	93.00-94.01	sec	401	KBytes	3.25	Mbits/sec	3.276	ms	144/555	(26%)
[ 5]	94.01-95.01	sec	0.00	Bytes	0.00	bits/sec	3.276	ms	0/0	(nan%)
[ 5]	95.01-96.00	sec	478	KBytes	3.95	Mbits/sec	1.564	ms	2863/3352	(85%)
[ 5]	96.00-97.00	sec	1.10	MBytes	9.22	Mbits/sec	0.222	ms	125/1278	(9.8%)
[ 5]	97.00-98.00	sec	715	KBytes	5.86	Mbits/sec	4.141	ms	215/947	(23%)
[ 5]	98.00-99.00	sec	577	KBytes	4.73	Mbits/sec	2.286	ms	398/989	(40%)
[ 5]	99.00-100.00	sec	578	KBytes	4.74	Mbits/sec	2.331	ms	448/1040	(43%)
[ 5]	100.00-101.00	sec	578	KBytes	4.73	Mbits/sec	2.462	ms	499/1091	(46%)
[ 5]	101.00-102.00	sec	573	KBytes	4.70	Mbits/sec	3.026	ms	548/1135	(48%)
[ 5]	102.00-103.00	sec	578	KBytes	4.74	Mbits/sec	2.870	ms	599/1191	(50%)
[ 5]	103.00-104.00	sec	573	KBytes	4.70	Mbits/sec	3.437	ms	648/1235	(52%)
[ 5]	104.00-105.00	sec	579	KBytes	4.74	Mbits/sec	1.547	ms	701/1294	(54%)
[ 5]	105.00-106.00	sec	575	KBytes	4.71	Mbits/sec	1.128	ms	531/1120	(47%)
[ 5]	106.00-107.00	sec	576	KBytes	4.71	Mbits/sec	4.534	ms	200/790	(25%)
[ 5]	107.00-108.00	sec	576	KBytes	4.71	Mbits/sec	2.273	ms	250/840	(30%)

[ 5]	83.00-84.00	sec	572	KBytes	4.69	Mbits/sec				
[ 5]	84.00-85.00	sec	572	KBytes	4.68	Mbits/sec				
[ 5]	85.00-86.00	sec	570	KBytes	4.67	Mbits/sec				
[ 5]	86.00-87.00	sec	569	KBytes	4.66	Mbits/sec				
[ 5]	87.00-88.00	sec	575	KBytes	4.72	Mbits/sec				
[ 5]	88.00-89.00	sec	575	KBytes	4.71	Mbits/sec				
[ 5]	89.00-90.00	sec	567	KBytes	4.65	Mbits/sec				
[ 5]	90.00-91.00	sec	566	KBytes	4.63	Mbits/sec				
[ 5]	91.00-92.00	sec	579	KBytes	4.74	Mbits/sec				
[ 5]	92.00-93.00	sec	394	KBytes	3.21	Mbits/sec				
[ 5]	93.00-94.00	sec	0.00	Bytes	0.00	bits/sec				
[ 5]	94.00-95.00	sec	0.00	Bytes	0.00	bits/sec				
[ 5]	95.00-96.00	sec	0.00	Bytes	0.00	bits/sec				
[ 5]	96.00-97.00	sec	418	KBytes	3.42	Mbits/sec				
[ 5]	97.00-98.00	sec	566	KBytes	4.64	Mbits/sec				
[ 5]	98.00-99.00	sec	573	KBytes	4.71	Mbits/sec				
[ 5]	99.00-100.00	sec	585	KBytes	4.79	Mbits/sec				
[ 5]	100.00-101.00	sec	577	KBytes	4.73	Mbits/sec				
[ 5]	101.00-102.00	sec	573	KBytes	4.70	Mbits/sec				
[ 5]	102.00-103.00	sec	579	KBytes	4.74	Mbits/sec				
[ 5]	103.00-104.00	sec	573	KBytes	4.69	Mbits/sec				
[ 5]	104.00-105.00	sec	577	KBytes	4.73	Mbits/sec				
[ 5]	105.00-106.00	sec	576	KBytes	4.73	Mbits/sec				
[ 5]	106.00-107.00	sec	576	KBytes	4.72	Mbits/sec				

## Resultados do Iperf3

# Conclusão

# Conclusão

- O handover Wi-Fi impacta significativamente o desempenho de TCP e UDP.
- TCP sofre mais com perdas e latência, enquanto UDP é mais resiliente.
- A mobilidade e a distância entre APs afetam a taxa de perda de pacotes