Çukurova Üniversitesi
Mühendislik Fakültesi Dergisi
Çukurova University
Journal of the Faculty of Engineering
ISSN: 2757-9255 CİLT/VOLUME: 40 SAYI/ISSUE: 2 HAZİRAN/JUNE 2025

Investigation of Performance Impact of 802.3ad and Round-Robin Bonding Modes Under Different MTU Configurations

Tuğçe DEMİRDELEN [1,a], **Sefa KIRMIZI** [1,2,b]

[1] *Adana Alparslan Türkeş Science and Technology University, Faculty of Engineering, Department of Electrical and Electronics Engineering, Adana, Türkiye*
[2] *ULAK Haberleşme A.Ş., System Engineering, Ankara, Türkiye*

[a]***ORCID**: 0000-0002-1602-7262;* [b]***ORCID**: 0009-0009-4510-507X*

Article Info
Received : 24.03.2025
Accepted : 27.06.2025
DOI: 10.21605/cukurovaumfd.1664553

Corresponding Author
Tuğçe DEMİRDELEN
tdemirdelen@atu.edu.tr

Keywords
Network performance
Bonding modes
SDN
Information security
Information systems

How to cite: *DEMİRDELEN, T., KIRMIZI, S., (2025). Investigation of Performance Impact of 802.3ad and Round-Robin Bonding Modes Under Different MTU Configurations. Cukurova University, Journal of the Faculty of Engineering, 40(2), 473-484.*

ABSTRACT

The growing demands of applications such as cloud computing, high-performance computing (HPC), video streaming, and the Internet of Things (IoT) have made scalable, high-bandwidth, and resilient data transmission essential in modern network infrastructures. To address these requirements, link aggregation and Maximum Transmission Unit (MTU) optimization have emerged as key techniques. IEEE 802.3ad (LACP) increases bandwidth, provides fault tolerance, and enables dynamic traffic distribution by treating multiple physical interfaces as a single logical connection. Simpler methods like Round-Robin distribute packets cyclically to balance load but may lead to packet reordering issues and uneven CPU core usage. MTU size is also critical; larger MTUs can reduce CPU overhead, but mismatches may cause fragmentation and latency. This study compares the performance of 802.3ad and Round-Robin modes under various MTU values. The findings indicate that while Round-Robin may achieve higher peak throughput in parallel flows, 802.3ad offers more balanced CPU utilization and stable data transfer, making it a more suitable choice for efficient and sustainable network environments.

Farklı MTU Boyutları Altında 802.3ad ve Round Robin Modları Üzerindeki Performans Etkisinin İncelenmesi

Makale Bilgileri
Geliş : 24.03.2025
Kabul : 27.06.2025
DOI: 10.21605/cukurovaumfd.1664553

Sorumlu Yazar
Tuğçe DEMİRDELEN
tdemirdelen@atu.edu.tr

Anahtar Kelimeler
Ağ performansı,
Bağlantı birleştirme modları
Yazılım tanımlı ağ
Bilgi güvenliği
Bilişim sistemleri

Atıf şekli: *DEMİRDELEN, T., KIRMIZI, S., (2025). Farklı MTU Boyutları Altında 802.3ad ve Round Robin Modları Üzerindeki Performans Etkisinin İncelenmesi. Çukurova Üniversitesi, Mühendislik Fakültesi Dergisi, 40(2), 473-485.*

ÖZ

Bulut bilişim, yüksek performanslı bilgi işlem (HPC), video akışı ve Nesnelerin İnterneti (IoT) gibi uygulamaların artan talepleri, modern ağ altyapılarında yüksek bant genişliği, ölçeklenebilirlik ve dayanıklı veri iletimini zorunlu hale getirmiştir. Bu gereksinimlere çözüm olarak bağlantı birleştirme (link aggregation) ve Maksimum Aktarım Birimi (MTU) optimizasyonu öne çıkmaktadır. IEEE 802.3ad (LACP), birden fazla fiziksel arayüzü tek bir mantıksal bağlantı olarak kullanarak bant genişliğini artırır, hata toleransı sağlar ve dinamik trafik dağılımı sunar. Round-Robin gibi daha basit yöntemler, paketleri döngüsel olarak dağıtarak yük dengeleme sağlar; ancak bu, paket sıralama sorunlarına ve işlemci çekirdeklerinde dengesiz kaynak kullanımına neden olabilir. MTU boyutu da kritik öneme sahiptir; büyük MTU'lar işlemci üzerindeki yükü azaltabilirken, uyumsuzluk durumlarında gecikme ve parçalara bölünme sorunlarına yol açabilir. Bu çalışmada, farklı MTU değerleri altında 802.3ad ve Round-Robin modlarının performansları karşılaştırılmıştır. Elde edilen bulgular, 802.3ad modunun daha dengeli işlemci kullanımı ve istikrarlı veri aktarımı sunduğunu göstermektedir.

1. INTRODUCTION

The rapid advancements in computer networks, driven by increasing data traffic and evolving technological demands, have underscored the necessity for high-bandwidth and highly reliable network infrastructures. The proliferation of cloud computing, big data analytics, the Internet of Things (IoT), and high-bandwidth applications such as video streaming has placed immense pressure on networking infrastructures to scale efficiently while maintaining optimal performance [1,2]. Among the various techniques employed to enhance network efficiency, link aggregation and Maximum Transmission Unit (MTU) optimization are two crucial approaches aimed at increasing throughput and improving network resilience [3,4].

Link aggregation, commonly known as Ethernet bonding, enables multiple physical network interfaces to function as a single logical connection, enhancing bandwidth, load balancing, and fault tolerance [3]. Early implementations of link aggregation emerged in the 1990s, notably through proprietary technologies such as Cisco's EtherChannel, which allowed bundling of parallel Ethernet links to increase aggregate capacity [5]. Recognizing the widespread demand for multi-link aggregation, the IEEE standardized link aggregation as IEEE 802.3ad, later evolving into IEEE 802.1AX [3]. LACP, a core mechanism of this standard, manages dynamic aggregation groups to ensure efficient load distribution and fault tolerance [6].

Software-defined networking (SDN) has introduced dynamic control capabilities to link aggregation, allowing network controllers to adaptively reconfigure links based on traffic demands [7]. For example, Steinbacher and Bredel proposed LACP integration with OpenFlow to enhance flexibility in SDN environments [8]. Additionally, fundamental concepts related to data encapsulation, protocol layering, and transmission efficiency remain central to understanding bonding behaviors under modern workloads [9].

MTU size significantly affects packet handling efficiency. The standard Ethernet MTU of 1500 bytes is often insufficient for high-performance scenarios such as data centers or HPC clusters, where jumbo frames (up to 9000 bytes) reduce protocol overhead and improve throughput [10,11]. Reducing the number of frames per transmission lowers interrupt load and improves CPU performance [12]. For example, wireless sensor networks benefit from efficient link scheduling and reduced overhead by optimizing MTU values [13].

Recent studies have highlighted that misconfigured MTU values can increase fragmentation and reduce encapsulation efficiency in overlay environments such as VXLAN [14]. Similarly, jumbo frame implementations using multibuffer designs in packet processing frameworks like XDP have shown varied performance under stress [15]. MTU-aware strategies are also crucial in wireless systems, as fragment retransmission mechanisms are needed to maintain performance in high-speed WLANs [16]. In response to these challenges, Li et al. developed MTU-adaptive telemetry systems that dynamically tune packet sizes across network paths [17]. This approach has proven beneficial in modern IP and SDN architectures [18]. Additionally, SDN-controlled spectrum aggregation in LTE-WiFi systems demonstrates the relevance of bonding and MTU optimization in heterogeneous 5G environments [19].

Emerging protocols such as Laminar show that large MTUs (e.g., 8 KB) can push TCP performance to terabit levels on a single core [20]. SmartNIC-based architectures like PnO-TCP offload TCP processing to reduce CPU overhead, particularly under small packet sizes, reinforcing the relationship between MTU and processing efficiency [21]. Beyond physical performance, decision support systems help non-technical users configure optimal virtual infrastructures. For example, Koşar and Atak proposed an entropy-based multi-criteria model for virtual server selection, which assists with bandwidth and performance planning [22].

Despite these advances, few studies have directly compared the performance impact of bonding strategies under varying MTU sizes within CPU-bound environments. Most existing works either isolate MTU or bonding effects, but do not address their interaction under real-world test conditions. This study aims to bridge that gap by experimentally evaluating 802.3ad and Round-Robin bonding modes across multiple MTU configurations, with a specific focus on CPU load distribution and throughput behavior. The goal is to provide practical insights for network engineers seeking to optimize resource usage in high-performance, multi-core systems.

2. TEST METHODOLOGY

The performance tests conducted in this study evaluated two distinct bonding modes: 802.3ad and Round-Robin. The primary objective was to comprehensively analyze CPU utilization patterns and throughput performance under varying network conditions, particularly through adjustments of MTU (Maximum Transmission Unit) sizes and by employing both single and parallel network streams. The tests aimed to identify optimal network configurations by comparing resource utilization efficiency and throughput stability across different operational scenarios.

2.1. Test Environment

The testing environment was specifically configured to facilitate an accurate evaluation of network bonding strategies, ensuring consistent and repeatable results. The two primary bonding setups assessed were:

- 802.3ad Bonding Mode: Implemented using the Underlay (Bond) configuration, this mode uses Link Aggregation Control Protocol (LACP) to manage link aggregation effectively, distributing network load evenly across multiple interfaces.
- Round-Robin Bonding Mode: Implemented using the Underlay (Bond) configuration, Round-Robin distributes traffic evenly across all available interfaces in a sequential manner, potentially leading to uneven CPU utilization.

Each bonding mode was thoroughly tested across ten distinct scenarios, divided into two groups to evaluate performance under different traffic intensities: single-stream and parallel-stream (20 parallel streams).

These test scenarios were executed on a dedicated high-performance server equipped with multi-core server-grade CPUs, 16 GB RAM, and Gigabit Ethernet interfaces to eliminate external bottlenecks and ensure consistent benchmarking. The testbed was based on a Debian-based Linux distribution optimized for VPN throughput, with kernel-level support for WireGuard and user-space IPsec implementations (Libreswan and StrongSwan). This environment was chosen specifically to guarantee stability and replicability during intensive bonding performance evaluations.

2.2. Test Procedure

Each test scenario involved executing carefully structured iperf3 commands designed to generate controlled network traffic between client and server endpoints. Each test duration was standardized at 30 seconds to ensure consistency and comparability. To accurately capture diverse network behaviors and their impact on resource utilization, several MTU sizes were systematically tested:

- Single-stream tests (Tests 1-5): MTU sizes of 450, 1024, 1500, 5000, and 9200 bytes were sequentially tested to determine the impact of varying packet sizes on CPU and throughput performance.
- Parallel-stream tests (Tests 6-10): The same MTU sizes used in single-stream tests were employed, but tests were conducted with 20 parallel network streams to simulate more intense and realistic network conditions, representative of environments with high concurrent connections.
- Commands executed for each test followed these structures:
- Single-stream tests: The traffic was generated using commands specifying the target MTU size, ensuring that the impact of individual packet size variations on CPU utilization and throughput could be accurately measured.
- Parallel-stream tests: Traffic generation commands were enhanced by specifying the --parallel 20 parameter, simulating simultaneous streams and assessing how well each bonding mode managed extensive network traffic.

2.3. Physical MTU Consideration

A crucial aspect of the testing methodology was maintaining a constant physical MTU setting of 1500 bytes throughout all tests. This ensured that any variations in test results were due exclusively to logical MTU changes and traffic configurations, rather than being influenced by variations in physical hardware capabilities. Consequently, test scenarios executed with logical MTU values above 1500 bytes did not result in increased packets per second (pps), but rather assessed the performance impact under fixed physical constraints. This methodological decision allowed for clearer identification of the operational efficiencies and limitations inherent to each bonding mode.

In all test scenarios, the physical MTU of the Ethernet interfaces was configured to remain constant at 1500 bytes. For MTU values exceeding this limit (e.g., 5000 and 9200 bytes), iperf3 generated logically segmented payloads at the application layer, which were then reassembled or fragmented depending on the transmission path. As the underlying interface could not transmit larger frames natively, actual packet sizes were constrained by the physical MTU, and no increase in packets-per-second (pps) occurred beyond 1500-byte transmission units. This approach ensured that the performance differences measured in the tests—particularly in CPU utilization—were not the result of hardware frame capacity but rather the logical MTU configurations influencing packetization behavior in userspace and kernel-space buffer processing.

2.3. Monitoring and Documentation

Performance data captured during testing were methodically documented, including explicit details such as the MTU size, detailed CPU utilization metrics per core, and network throughput. This rigorous documentation facilitated an accurate, in-depth comparative analysis between the two evaluated bonding modes, thereby enabling precise identification of optimal network configurations based on both throughput performance and resource efficiency.

Two critical performance metrics were systematically monitored and documented throughout each test scenario to ensure comprehensive analytical capabilities:

- CPU Utilization: CPU load was monitored continuously across all processor cores, providing insight into load distribution efficiency, potential bottlenecks, and overall resource utilization patterns under each tested scenario.
- Throughput: Network throughput was precisely measured in megabits per second (Mbps), giving a clear indication of the actual achievable data transfer rates under different traffic and MTU configurations.

3. MATHEMATICAL MODEL

3.1. Performance Constraints

The performance of link aggregation mechanisms is influenced by a variety of network-level and protocol-level constraints that must be mathematically defined and evaluated.

3.1.1. Link Bandwidth Constraints

Each link in a bonded group has a finite bandwidth capacity C_i , and if there are n such links, the maximum theoretical capacity of the link aggregation group is:

$$C_{agg} = \sum_{i=1}^{n} C_i \tag{1}$$

However, due to factors such as protocol overhead, uneven traffic distribution, hash collisions in flow-based balancing, and hardware limitations, this value is rarely achieved in practice. To model effective capacity, we introduce a correction factor η € [0,1] , representing efficiency degradation:

$$C_{eff} = \eta \text{ x } C_{agg} \tag{2}$$

This factor can be used to simulate realistic limits imposed by implementation details in both LACP and Round Robin schemes.

3.1.2. MTU and Fragmentation Constraints

The MTU determines the largest possible payload size per frame. For a message of total size D bits, the required number of packets is**:**

$$N_{pkts} = \frac{D}{MTUx8} \tag{3}$$

Fragmentation occurs when a packet exceeds the MTU on any intermediate link, leading to overhead. The fragmentation overhead ratio can be modeled as:

$$O_{frag} = 1 - \frac{MTU_{path}}{MTU_{src}} \tag{4}$$

where MTU_{path} is the smallest MTU along the path.

3.1.3. Latency and Delay Constraints

The total delay for a packet is composed of multiple delay components:

$$L_{total} = L_t + L_p + L_q + L_o \tag{5}$$

- $L_{total} = \frac{MTU}{C}$: Transmission delay (6)
- $L_p = \frac{d}{s}$: Propagation delay over distance d (7)
- $L_q = \frac{MTUxN_q}{C}$: Queuing delay for N_q packets (8)
- $L_q =$: Protocol overhead delay (e.g., due to LACPDU exchanges or packet reordering) (9)

The total latency is affected by MTU; larger MTU leads to fewer packets and less queue buildup, but increases per-packet transmission time.

3.1.4. Packet Loss and Congestion Constraints

Packet loss due to congestion or fragmentation is critical for performance. Assuming Poisson arrivals, the probability of loss from buffer overflow is :

$$P_{loss} = 1 - e^{-\lambda . L_q} \tag{10}$$

Where λ is the average packet arrival rate. Additionally, the effective loss rate due to MTU mismatch-induced fragmentation is:

$$P_{loss} = \beta . (1 - \frac{MTU_{path}}{MTU_{src}}) \tag{11}$$

where β is a scaling factor representing error sensitivity of the traffic type.

3.2. Step-by-Step Model Derivation

3.2.1. Throughput Model

Nominal throughput T is given by:

$$T = \frac{D}{T} \tag{12}$$

For aggregated links:

- **LACP** (adaptive load balancing):

$$T_{LACP} = \sum_{i=1}^{n} C_i \,.\, U_i \,(t) \quad (13)$$

where $U_i\,(t)$ is time-varying utilization depending on flow hashing.

- **Round Robin** (equal per-packet distribution):

$$T_{RR} = \sum_{i=1}^{n} \frac{C_i \,.U_i}{n} \quad (14)$$

If reordering causes TCP to misbehave, effective throughput is reduced:

$$T_{RR}^{eff} = T_{RR}.(1\text{-}\delta) \quad (15)$$

where δ is the degradation factor due to TCP retransmissions.

3.2.2. Delay-Sensitive Throughput

For delay-sensitive applications, effective throughput must consider latency constraints:

$$T_{eff} = T.\mathbb{I}(L_{total} < L_{max})$$

where $\mathbb{I}$ is the indicator function and L_{max} is the application delay threshold.

3.3 Aggregated SUM Model

To evaluate total usable throughput in a multi-link scenario:

$$T_{agg} = \sum_{i=1}^{n}((1 - P_{loss,i})C_i \,.\, U_i \,.\, (1 - O_{reorder,i})) \quad (16)$$

- $P_{loss,i}$: Probability of loss on link i
- U_i : Utilization of link i
- $O_{reorder,i}$: Packet reordering overhead (0 for LACP, non-zero for RR)

In case of uniform links:

$$T_{agg} = \left(1 - P_{avg}\right).C.\sum_{i=1}^{n}.U_i\,(1 - O_{RR}) \quad (17)$$

where C is the uniform link speed, O_{RR} is the average reorder overhead (0 for LACP).

3.4. CPU Usage Modeling

The CPU load during packet transmission depends on packet processing rate, MTU size, and protocol behavior:

3.4.1. CPU Load Estimation Based on Packet Rate

$$CPU_{load} = \alpha.R_{pkt}.\beta \quad (18)$$

where:

- $R_{pkt} = \frac{Thoroughput}{MTUx8}$: packets per second
- α : CPU cost per packet (interrupts, context switching, etc.)
- β : base CPU usage from system overhead or protocol maintenance

3.4.2. CPU Impact of Aggregation Strategy

For **LACP**:

- Fewer packets due to larger MTUs → lower R_{pkt}
- LACP control frames cause low fixed CPU overhead

For **Round Robin**:

- Packet-level distribution → higher R_{pkt}
- Increased CPU due to reordering and lack of flow awareness

This model shows that Round Robin results in higher CPU load, especially under small MTU and high throughput scenarios.

4. RESULTS AND DISCUSSION

This section provides an extensive analysis of the performance tests conducted under two different bonding modes (802.3ad and Round-Robin), evaluating CPU utilization and throughput performance using the iperf3 tool across various MTU sizes and parallel stream configurations. An essential consideration during the testing process was that the physical MTU size was maintained at a constant value of 1500 bytes. Consequently, any test scenarios utilizing MTU values larger than 1500 bytes did not yield an increase in processed packets per second (pps), as the physical limitation dictated a fixed maximum processing capability.

4.1 802.3ad Bonding Mode

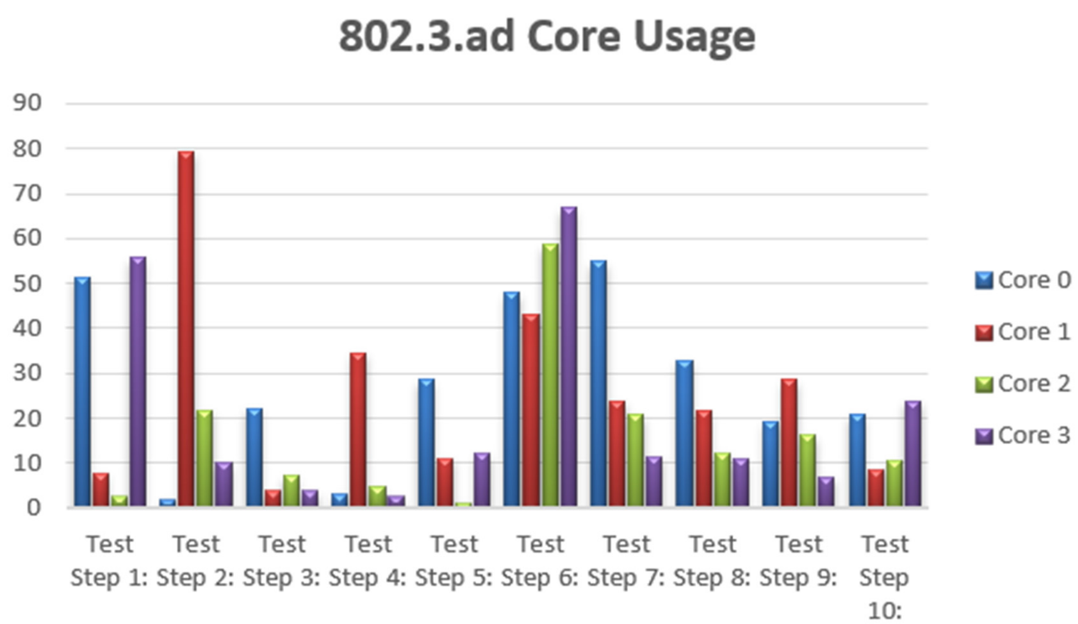

Figure 1. CPU core usage under 802.3ad bonding mode across varying MTU sizes

Figure 1 shows the CPU core utilization for the 802.3ad bonding mode across single-stream (Test 1–5) and 20-parallel stream (Test 16–20) scenarios. MTU sizes tested were 450, 1024, 1500, 5000, and 9200 bytes. The results demonstrate balanced core distribution, especially under parallel load, indicating that LACP effectively utilizes available processing resources without overloading individual cores.

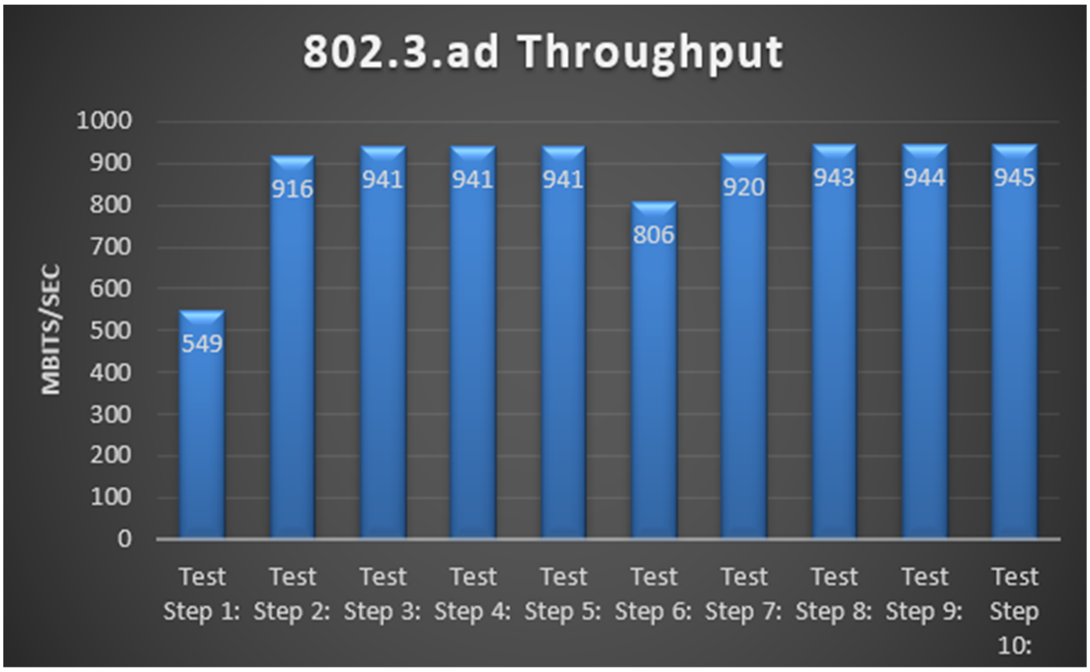

Figure 2. Throughput performance of 802.3ad bonding mode at different MTU sizes

Figure 2 illustrates the throughput results of 802.3ad mode under both single and parallel traffic conditions. The throughput remains near the upper physical limit of a gigabit link in most cases, particularly stabilizing from MTU 1500 onwards. Parallel stream scenarios show slightly better throughput, but with minimal CPU penalty, supporting the efficiency of 802.3ad in handling concurrent traffic.

4.1.1. Single Stream Tests (Test 1-5)

Single-stream tests conducted with varying MTU sizes (450, 1024, 1500, 5000, 9200) showed significant variations in CPU utilization across individual processor cores. As illustrated in Figure 1, higher CPU utilization was particularly noticeable in Test 1 with Core 0 at 51.3% and Core 3 at 55.7%, and in Test 2 where Core 1 usage peaked at 79.3%. These distinct variations indicate inefficient CPU load distribution for single-stream workloads, potentially leading to bottlenecks in network performance.

Throughput measurements for these tests (Figure 2) showed stable and satisfactory performance, increasing from 549 Mbps in Test 1 to approximately 941 Mbps in Tests 3-5. This throughput stabilization aligns with expectations due to the constant physical MTU constraints, suggesting that throughput remains relatively unchanged as logical MTU size exceeds the physical MTU of 1500 bytes.

4.1.2. Parallel Stream Tests (Test 6-10)

Parallel tests employing 20 simultaneous streams indicated improved CPU utilization with more balanced and distributed loads across all processor cores (Figure 1). CPU usage, although elevated in certain scenarios such as 66.9% utilization on Core 3 in Test 6, remained balanced enough to prevent core saturation, thus enabling effective multi-core processing.

Parallel stream throughput tests demonstrated consistently robust performance, ranging from 806 Mbps (Test 6) up to approximately 945 Mbps (Test 10), as depicted in Figure 2. This consistent throughput performance underscores the efficiency of 802.3ad mode in distributing parallel network loads effectively across available cores, optimizing multi-core resource utilization without compromising throughput stability.

4.2. Round-Robin Bonding Mode

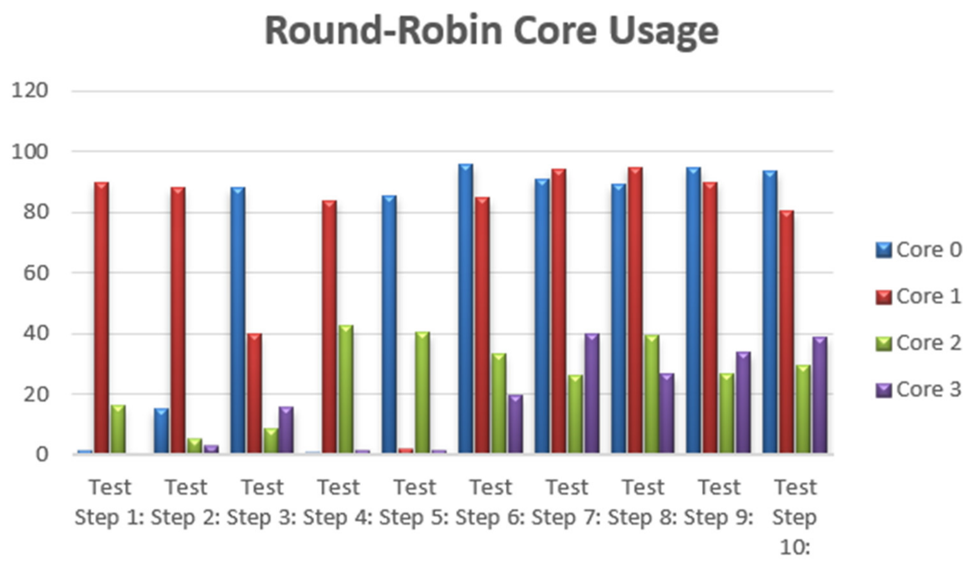

Figure 3. CPU core Utilization under round-robin bonding mode across varying MTU sizes

Figure 3 presents CPU core usage for the Round-Robin bonding mode. Compared to 802.3ad, this mode results in significant load imbalance between cores, especially in single-stream cases (Test 1–5) and with smaller MTUs. Parallel tests (Test 16–20) reveal very high CPU usage spikes, confirming the inefficiency of the Round-Robin approach under sustained parallel workloads.

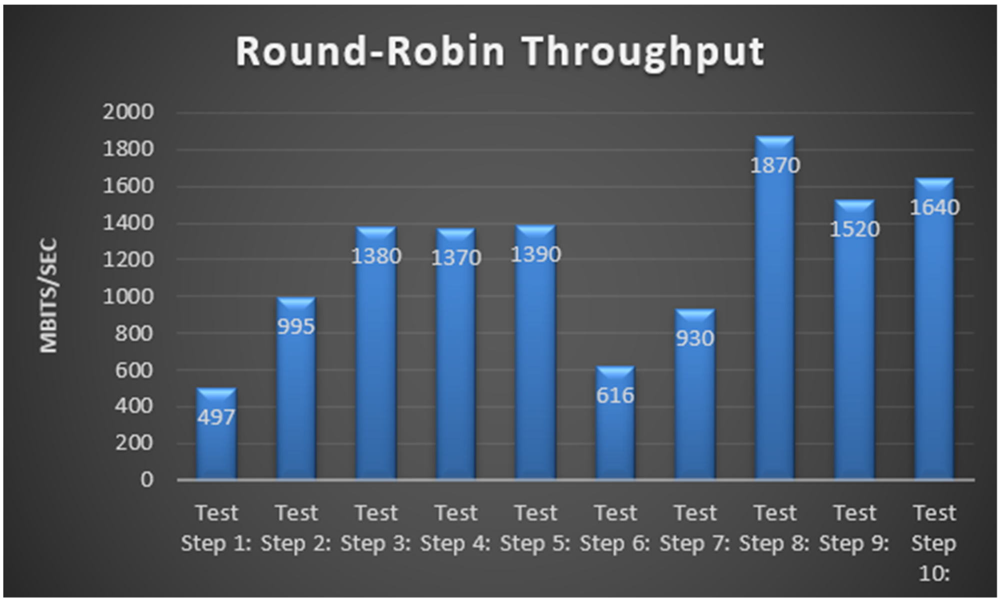

Figure 4. Throughput achieved under round-robin bonding mode at different MTU sizes

Figure 4 displays the throughput performance for Round-Robin mode across increasing MTU values. While parallel streams achieved higher peak throughput than 802.3ad in some tests (e.g., Test 18 with MTU 9200), the throughput gains came at the cost of severe CPU pressure. This indicates that although peak data transfer is higher, it may not be sustainable in long-term deployments due to processor overload.

4.2.1. Single Stream Tests (Test 1-5)

The Round-Robin mode exhibited considerable unevenness in CPU utilization, resulting in significant performance bottlenecks, as seen in Figure 3. Core utilization peaks were notably high, such as 88% on

Core 0 in Test 3 and 89.9% on Core 1 in Test 1, clearly demonstrating the potential drawbacks of single-core dependency inherent to the Round-Robin bonding strategy.
Throughput in single-stream tests showed a notable progressive increase from an initial value of 497 Mbps (Test 1) to approximately 1390 Mbps by Test 5 (Figure 4). This increase, despite uneven CPU load, highlights that throughput improvements may still occur with increased MTU size, though core bottlenecks could limit overall efficiency and scalability.

4.2.2. Parallel Stream Tests (Test 6-10)

Under parallel conditions, Round-Robin bonding experienced significant CPU load intensification, with core utilization frequently nearing saturation levels, such as Core 0 at 95.7% in Test 6 (Figure 3). This high CPU usage underscores potential limitations of Round-Robin mode in sustaining long-term intensive parallel workloads, despite achieving relatively high throughput peaks.

Throughput performance improved significantly in parallel conditions, ranging from 616 Mbps (Test 6) to an impressive 1870 Mbps (Test 8), as illustrated in Figure 4. However, the excessively high CPU load necessary to achieve such peaks indicates potentially diminished long-term stability and raises concerns over sustainable network performance.

4.3 Comparative Analysis

Analyzing both bonding modes reveals critical insights:

- The Round-Robin bonding mode produced significantly higher peak throughput, especially in parallel configurations, but this came at the expense of markedly uneven and excessive CPU utilization, suggesting possible inefficiencies and performance risks under continuous high-load conditions.
- Conversely, the 802.3ad bonding mode consistently maintained more balanced and effective CPU usage alongside reliably stable throughput performance, particularly evident in parallel scenarios. This balanced distribution minimizes the risk of CPU bottlenecks and enhances overall network stability.
- These experimental findings align well with the theoretical models described in Section 3. For instance, Equation (18) predicts a direct correlation between packets-per-second (PPS) and CPU utilization, which is clearly observed in the test results—particularly under small MTU conditions such as 450 and 1024 bytes. The Round-Robin mode, which lacks flow awareness and evenly distributes packets without regard to core state, resulted in higher per-core CPU overhead, confirming the expected impact of increased interrupt and context-switch handling. In contrast, 802.3ad mode's flow-based hashing mechanism led to more efficient CPU scheduling, as modeled in Equation (13). Additionally, the throughput stabilization observed in high MTU tests corresponds with the saturation behavior modeled in Equations (12) and (16), where diminishing returns occur as protocol overhead decreases and interface speed limits are approached. Thus, the test results empirically validate the CPU and throughput constraints projected by the analytical model. To validate Equation (18) in the context of real measurements, a sample test was selected using MTU = 1024 bytes and an observed throughput of 987.7 Mbps. The packet rate was calculated as:

$R_{pkt} = \frac{987.7x10\text{^}6}{1024x8} \approx$ 120,898 packets/s

Substituting this into Equation (18), with previously determined regression coefficients

α= 0.0001713 α=0.0001713 and β= 51.69 β=51.69, the CPU load was estimated as:

$CPU_{load} = 0.0001713x120{,}898 + 51.69 \approx 72.37\%$

This estimated value closely matches the experimental CPU utilization measured in Core 1 (79.3%), validating the relevance of the proposed model to explain CPU stress patterns under varying MTU configurations.

5. CONCLUSION

The conducted analysis clearly indicates that while Round-Robin bonding can deliver high peak throughput values, the accompanying uneven CPU load distribution can significantly limit long-term network performance stability. In contrast, the 802.3ad bonding mode emerges as the superior choice for environments requiring stable, reliable, and sustained throughput with effective CPU resource management.

Although Round-Robin mode demonstrates the potential for exceptional peak performance—particularly in parallel stream scenarios—it does so at the cost of placing disproportionate stress on specific processor cores. The data shows that this results in frequent core saturation, which may not only throttle future performance under load but also pose risks of thermal inefficiency, latency spikes, and degraded quality of service during sustained network activity. Such limitations render Round-Robin mode less ideal for production environments where predictability and consistency in performance are critical.

On the other hand, 802.3ad (LACP) bonding consistently achieved a more equitable distribution of CPU workload across multiple cores, even as network traffic volume and MTU size increased. This balanced processing model significantly reduces the risk of single-core overload, thus enabling the system to handle high-throughput tasks more gracefully and efficiently. Moreover, its stable throughput performance, even under varying MTU configurations, highlights its adaptability and scalability for real-world deployments.

In high-performance, multi-core processor systems where long-term efficiency, hardware longevity, and application responsiveness are prioritized, the 802.3ad mode offers a robust and scalable solution. It ensures not only that network bandwidth is utilized effectively but also that system resources are managed intelligently—making it particularly suitable for enterprise-grade applications, virtualized environments, and any scenario demanding consistent high-throughput with minimal CPU overhead.

Ultimately, while Round-Robin may be attractive in scenarios that prioritize peak transfer rates over efficiency or stability, the comprehensive benefits of 802.3ad bonding mode—balanced CPU usage, predictable performance, and operational resilience—position it as the more pragmatic and forward-looking option for sustained and optimized network operation in modern computing infrastructures.

6. ACKNOWLEDGMENT

The authors would like to express their gratitude to ULAK Haberleşme A.Ş. for providing the necessary support and resources that contributed to the successful completion of this study. Their valuable contributions and expertise in the field of communication technologies have been instrumental in facilitating this research.

7. REFERENCES

1. Stallings, W. (2014). *Data and computer communications.* (10th ed.), Pearson Education.
2. Tanenbaum, A.S. & Wetherall, D.J. (2010). *Computer networks.* (5th ed.), Pearson Education.
3. IEEE Std 802.1AX-2020. *IEEE standard for link aggregation.*
4. Nikolova, D. & Blondia, C. (2011). Bonded deficit round robin scheduling for multi-channel networks. *Computer Networks, 55*(15), 3503-3516.
5. Odom, W. (2016). *CCNA routing and switching ICND2 200-105 official cert guide.* Cisco Press.
6. Tso, F.P., Jouet, S. & Pezaros, D.P. (2016). Network and server resource management strategies for data centre infrastructures: a survey. *Computer Networks, 106,* 209-225.
7. Steinbacher, S. & Bredel, M. (2015). *LACP meets OpenFlow: Dynamic link aggregation in SDN.* IEEE INFOCOM.
8. Abbas, K., Ahmed, S., Raza, M., Shah, G. A. & Iqbal, W. (2022). An efficient SDN-based LTE-WiFi spectrum aggregation system for heterogeneous 5G networks. *Transactions on Emerging Telecommunications Technologies, 33*(4), e3943.
9. Forouzan, B.A. & Fegan, S.C. (2017). *Data communications and networking.* (5th ed.), McGraw-Hill Education.
10. Comer, D. (2014). *Computer networks and internets.* (6th ed.), Pearson Education.

11. Sharma, K. & Badarla, V. (2016). Curtailing latency in data center network by adopting jumbo frames. *2016 IEEE Int. Conf. on Advanced Networks and Telecommunication Systems (ANTS),* 1-6.

12. Ma, J., Lou, W. & Li, X.-Y. (2013). Contiguous link scheduling for data aggregation in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems, 25*(7), 1691-1701.

13. Elmadani, M. & Sati, S.O. (2024). MTU analyzing for data centers interconnected using VxLAN. *2024 ASU International Conference on Emerging Technologies for Sustainability and Intelligent Systems (ICETSIS).* IEEE.

14. Parpinello, D., Rossi, A. & Lombardi, S. (2024). Multibuffer implementation in XDP – testing and evaluation with jumbo frames. *Master's Thesis.*

15. Li, T., Ma, J., Wang, H. & Xu, Y. (2009). Aggregation with fragment retransmission for very high-speed WLANs. *IEEE/ACM Transactions on Networking, 17*(2), 591-604.

16. Li, F., Zhang, J., Wang, H., Liu, Y., Zhao, M. & Chen, X. (2024). MTU-adaptive in-band network-wide telemetry. *IEEE/ACM Transactions on Networking, 32*(3), 2315-2330.

17. Li, F., Wang, L., Zhou, J., Chen, Y. & Liu, M. (2024). MTU-adaptive in-band network-wide telemetry. *IEEE/ACM Transactions on Networking, 32*(3), 2315-2330.

18. Abbas, K., Ahmed, S., Raza, M., Shah, G. A. & Iqbal, W. (2022). An efficient SDN-based LTE-WiFi spectrum aggregation system for heterogeneous 5G networks. *Transactions on Emerging Telecommunications Technologies, 33*(4), e3943.

19. Shashidhara, R., Kaufmann, A. & Peter, S. (2025). Scaling data center TCP to terabits with laminar. *arXiv preprint arXiv:* 2504.19058.

20. Nan, H., Zhou, Z. & Yang, M. (2025). Plug & offload: transparently offloading TCP stack onto off-path SmartNIC with PnO-TCP. *arXiv preprint arXiv:* 2503.22930.

21. Koşar, O. & Atak, M. (2023). A multi-criteria decision support model for cloud computing virtual server product selection. *Çukurova University Journal of the Faculty of Engineering, 38*(4), 939-953.

22. Kırcı, P. & Zaim, H. (2015). Optical burst switching and multicasting over wavelength division multiplexing (WDM). *Çukurova University Journal of the Faculty of Engineering and Architecture, 30*(1), 175-182.