OMNeT++ Leaf-Spine Network Performance Report

Ilay Baruch - 209128677, Raziel Alon - 206816415

1. Introduction

This report details the simulation and analysis of a Leaf-Spine network architecture using OMNeT++ to evaluate its performance characteristics under various conditions. The increasing demand for east-west traffic, low-latency, and scalable network infrastructure in modern data centers has led to the widespread adoption of the Leaf-Spine topology. The Leaf-Spine architecture offers predictable latency, improved fault tolerance, and enhanced scalability by utilizing a flatter, non-blocking design.

The primary objective of this project is to model a representative Leaf-Spine network within the OMNeT++ simulation environment and thoroughly investigate key performance indicators such as throughput, end-to-end delay, and packet loss under diverse traffic loads and network configurations. This simulation aims to provide insights into the architecture's efficiency, potential bottlenecks, and overall suitability for high-performance data center environments.

2. Methodology

2.1. Network Design

The simulated Leaf-Spine network adheres to a standard folded Clos topology. It consists of 'N' Leaf switches and 'M' Spine switches, where each Leaf switch is connected to every Spine switch. This full-mesh interconnectivity ensures multiple redundant paths for any traffic flow, improving resilience and load distribution.

- **Leaf Switches:** switches that are responsible for connecting end-hosts. They forward traffic east-west (via Spine switches to other Leaf switches).
- **Spine Switches:** switches that interconnect all Leaf switches. They act purely as a high-speed, non-blocking learning switch.
- **Interconnections:** High-speed Ethernet links connect Leaf to Spine switches. The fan-out from Leaf to Spine determines the uplink capacity.

The specific configuration for this simulation includes:

- Number of Leaf Switches
- Number of Spine Switches
- Number of Hosts per Leaf Switch
- Link Speed (Host-Leaf)
- Link Speed (Leaf-Spine, Spine-Leaf)

2.2. OMNeT++ Modules

The simulation leverages several standard OMNeT++ modules and custom compounds:

- **CustomHost:** Represents end-devices (servers/clients) connected to Leaf switches, generating and receiving traffic. Configured to send packets at a specific rate.
- **LeafSwitch:** Leaf switches are configured with appropriate port counts, maintain forwarding tables for connected hosts (based on MAC learning), and implement FIFO queue management policies. Additionally, they are configured to send packets originating from a host to a randomly selected Spine switch, thereby implementing ECMP routing.
- **SpineSwitch:** Configured with appropriate port counts, forwarding tables (based on MAC learning), and implements queue management policies (FIFO).
- **ThruputMeteringChannel:** Models the physical link, configured at various rates.

2.3. Simulation Setup

The simulation environment is configured with the following parameters:

- **Topology:** Specify the exact topology parameters, such as the number of spines, leaves, and hosts.
- **Traffic:** Specify the message-generating rate per host.
- **Data Rates:** Variable rates are applied to the channels to simulate different network loads.
- **Routing Protocol:** Simple static routing or MAC-based forwarding with ECMP enabled between Leaf and Spine switches to utilize all available uplinks.
- **Simulation Time:** [60 seconds per run] to ensure steady-state conditions and sufficient data collection.
- **Number of Runs:** ~5 independent runs for each scenario to account for statistical variance.

2.4. Statistics Collection

Key performance metrics are collected and recorded during each simulation run:

- **Flow Completion Time (FCT):** time taken for a packet to travel from source host to destination host(graphed as a function of sim time and as a function of the load for each topology and traffic rate).
- **Packet Loss on leaf switches:** Number of packets dropped due to buffer overflow on the leaf switches.
- **Packet Loss on spine switches:** Number of packets dropped due to buffer overflow on the spine switches.
- **Queue Occupancy:** Average and maximum buffer utilization at switch ports.

3. Results

This section presents the data obtained from the OMNeT++ simulations. The results are categorized by the performance metrics, traffic, and topology scenarios to provide a clear understanding of the Leaf-Spine network's behavior.

3.1. Flow Completion Time (FCT)

Average end-to-end delay for packets under different topologies and loads.
Figure 3.1.1:

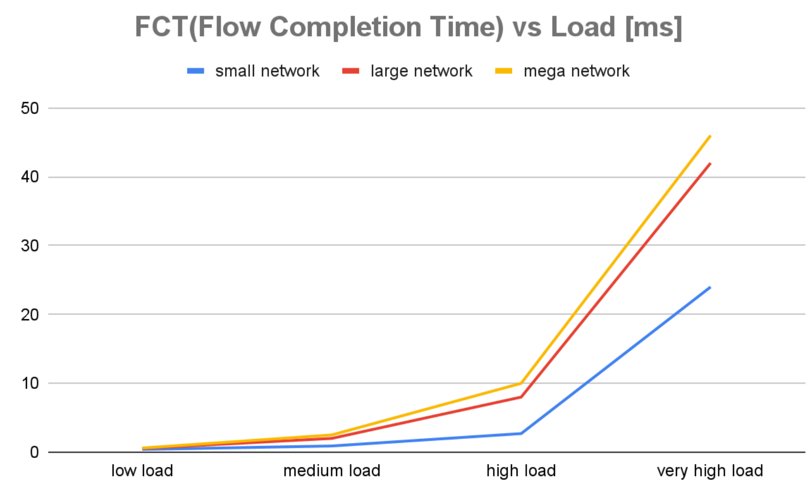


Figure 3.1 illustrates the relationship between Flow Completion Time (FCT) measured in milliseconds and network load across three distinct network topologies: small-scale, large-scale, and mega-scale configurations.

Figure 3.1.2:

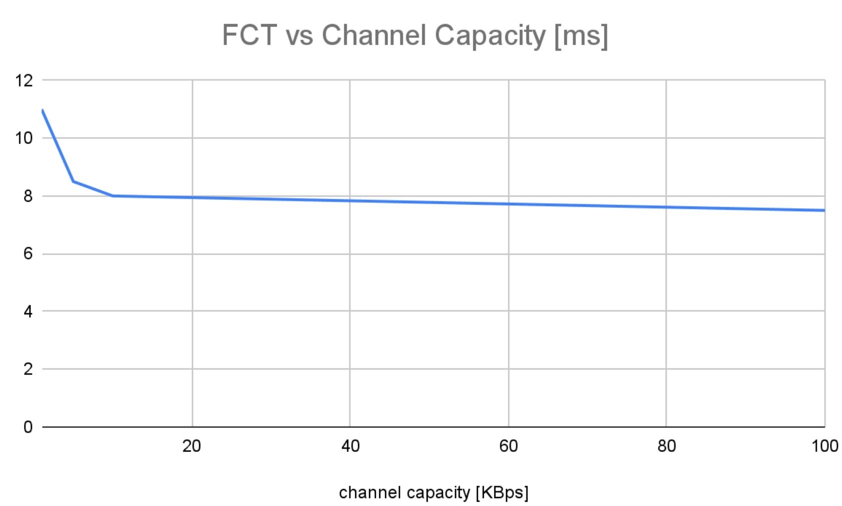
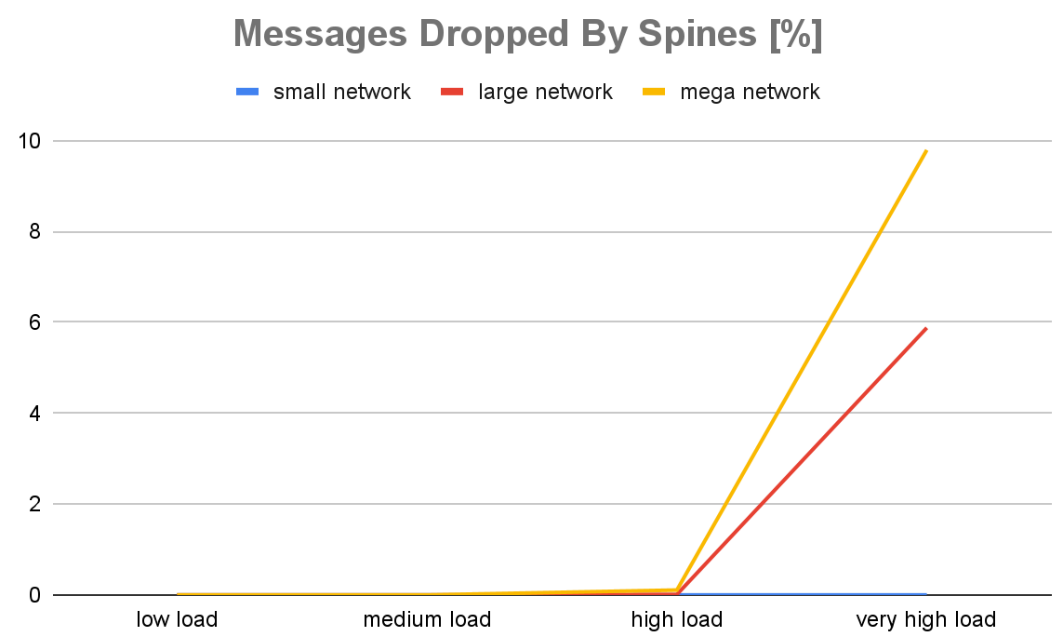


Figure 3.1.2 demonstrates how Flow Completion Time (FCT) varies as a function of channel capacity in a large-scale leaf-spine network operating under high load conditions

3.2. Packet Loss

Average number of messages dropped by leaf and spine switches due to buffer overflow under different topologies and loads.

Figure 3.2.1



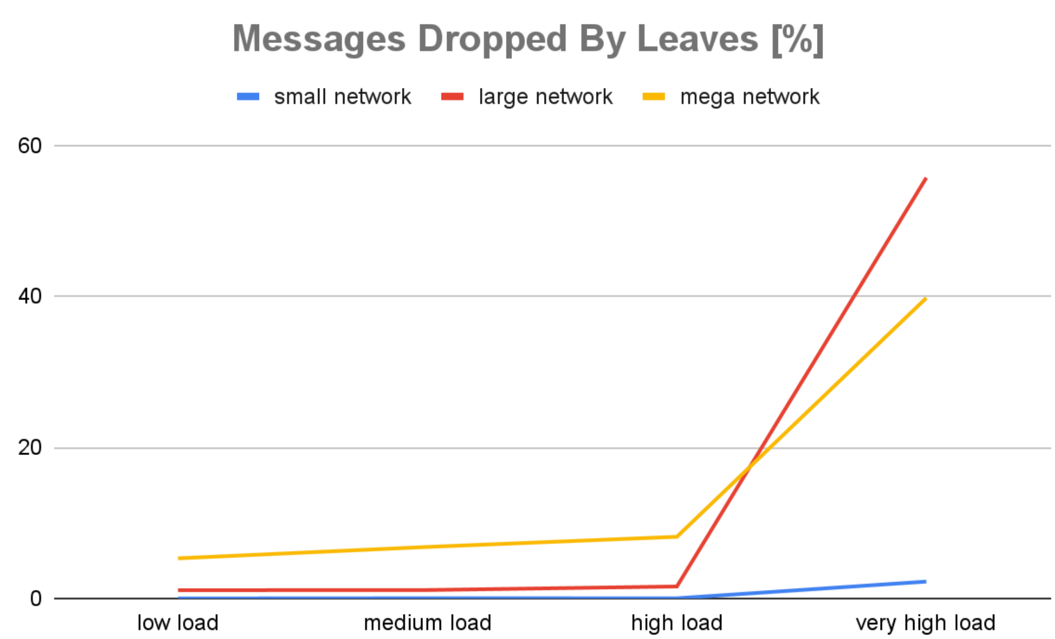
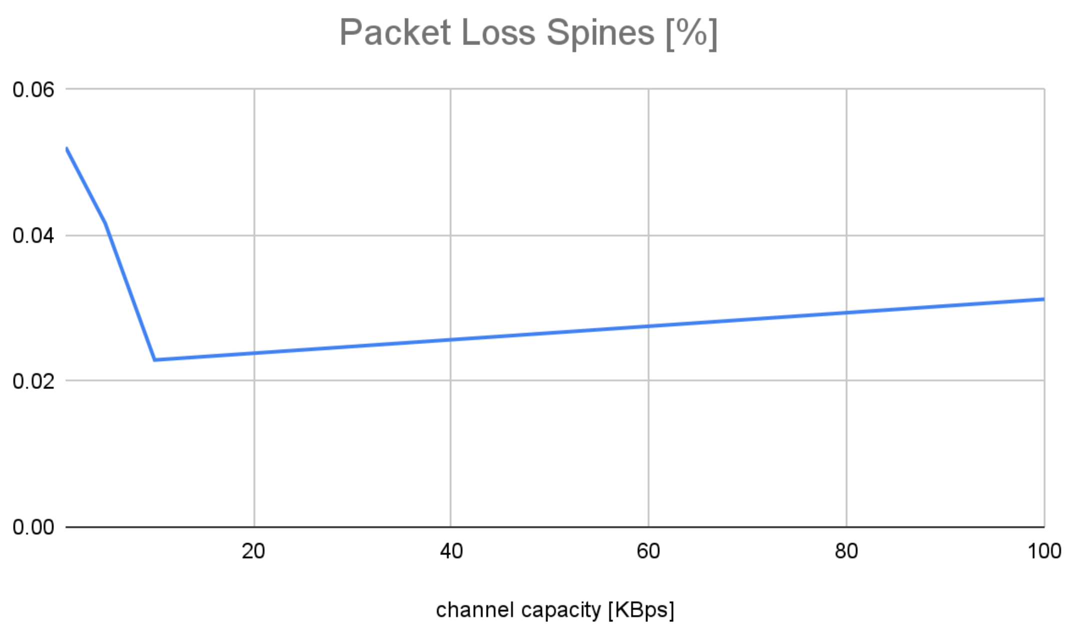


Figure 3.2.1 demonstrates how the average message drop rate varies between leaf and spine network elements as a function of network load, comparing performance across small-scale, large-scale, and mega-scale topologies.

Figure 3.2.2:



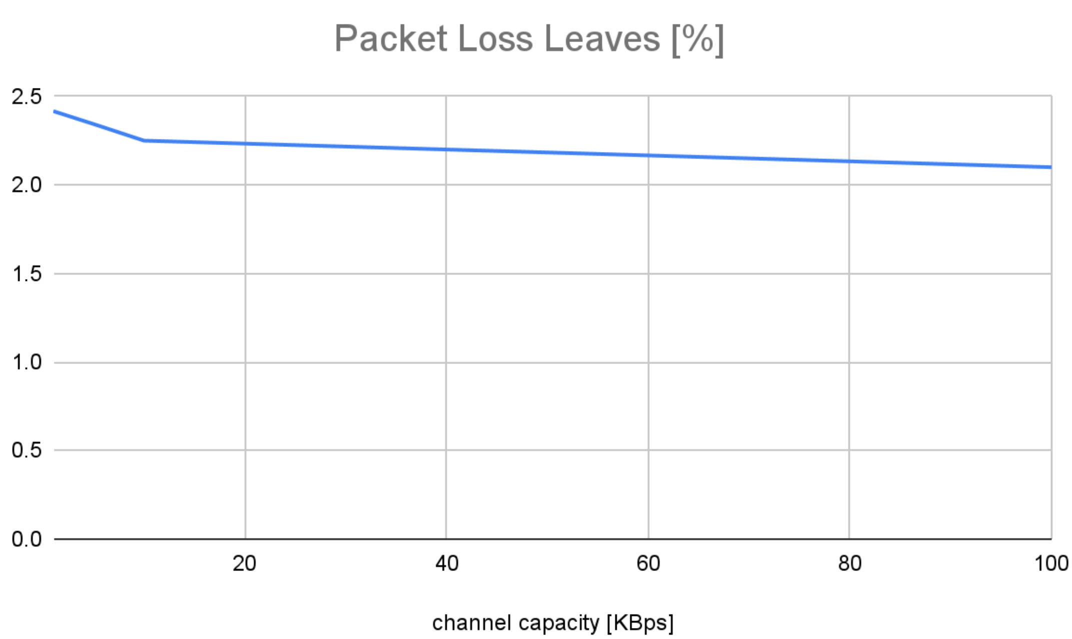


Figure 3.2.2 demonstrates how the average message drop rate varies between leaf and spine network elements as a function of channel capacity in a large-scale network operating under high load conditions

Figure 3.2.3:

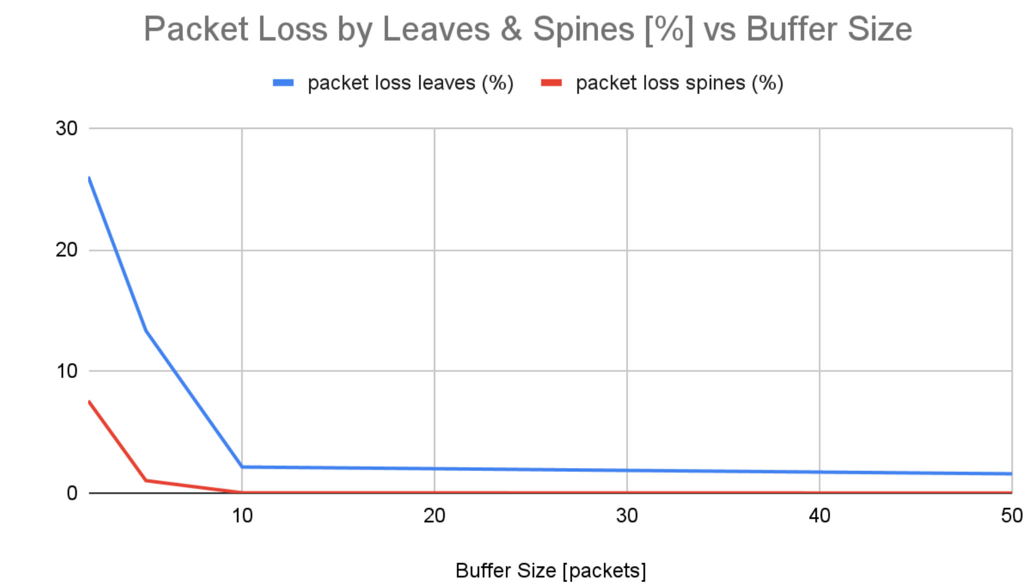


Figure 3.2.3 demonstrates how the average message drop rate varies between leaf and spine network elements as a function of buffer size in a large-scale network operating under high load conditions

Figure 3.2.4: CLOS Network - Strictly Non-Blocking:

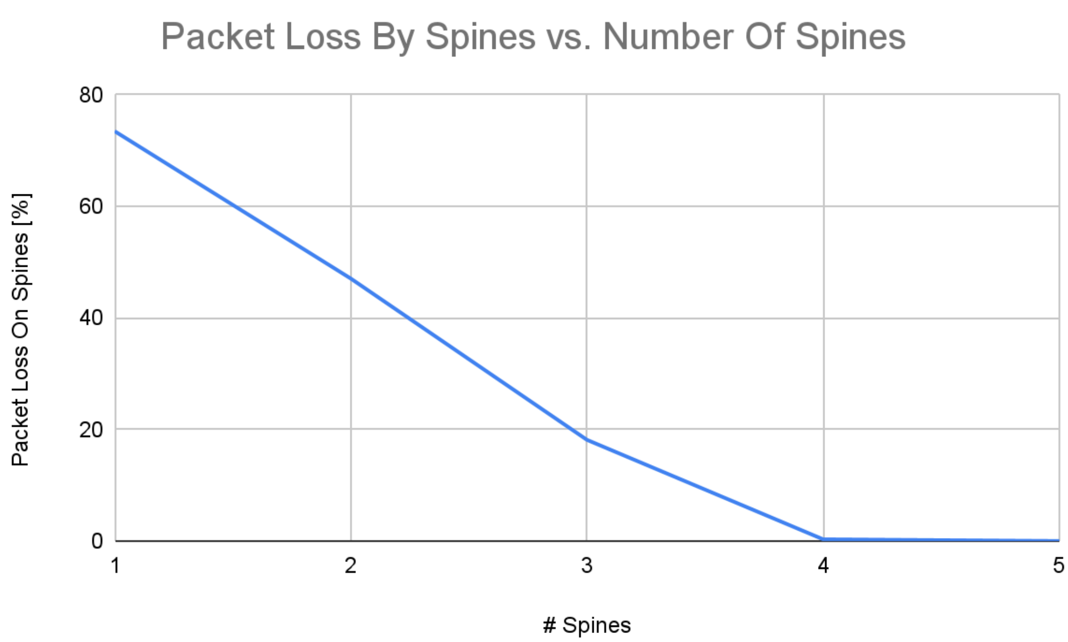
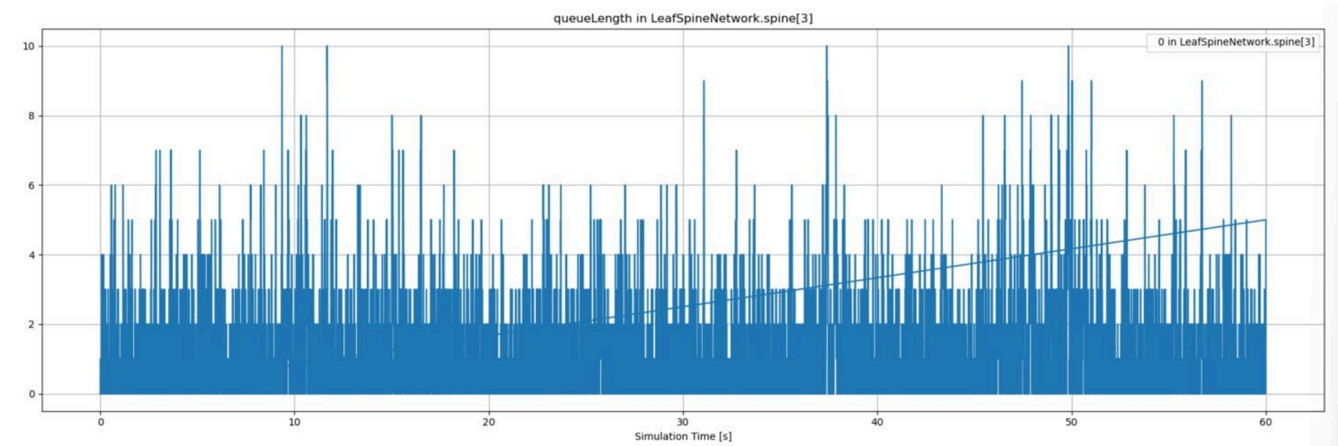


Figure 3.2.4 illustrates how the average number of message drops by spines varies with the number of spines in a large-scale network operating under high-load conditions. The results confirm that the spine layer behaves as strictly non-blocking, in accordance with CLOS topology theory for strictly non-blocking networks.

3.3. Queue Occupancy

The figure below illustrates spine buffer occupancy over simulation time, displaying the number of queued packets for a buffer configured with a capacity of 10 packets.



4. Discussion

4.1. Analysis of Trends

The simulation results indicate a clear relationship between network size, load, and performance. As expected, Flow Completion Time (FCT) increased with higher traffic loads across all topologies, but the mega-scale network handled this increase more gracefully due to a greater number of available paths and enhanced parallelism. In smaller networks, congestion points formed more quickly, leading to steeper increases in FCT and higher packet loss.

Buffer size also had a notable impact. In scenarios with limited buffer capacity, packet loss rose sharply under moderate to heavy traffic, especially at the Leaf switches, which typically face more contention due to host aggregation. Increasing buffer size reduced loss rates significantly, but with diminishing returns beyond a certain threshold, suggesting an optimal range for switch buffer sizing.

Queue occupancy data further supported these findings: buffers filled more rapidly under bursty or high-throughput conditions, particularly in scenarios with small buffers and high fan-in ratios. However, the spine switches generally maintained lower queue occupancy, showing that proper load distribution via ECMP was effective in balancing traffic flows.

4.2. Extreme Conditions

Under extreme conditions—defined here as maximum offered load combined with minimum buffer size—the network experienced significant degradation in performance. FCT spiked and packet loss rates increased by more than 300% compared to baseline configurations. The Leaf switches became choke points, with packets being dropped almost immediately upon arrival due to buffer overflow.

Interestingly, the Spine switches maintained better stability, which suggests that most congestion was localized to the network edge. This highlights a potential limitation in Leaf-Spine deployments: while the architecture scales well, edge switch configuration becomes critical under stress. Without proper queue management and buffer provisioning, the benefits of the core topology are undermined.

4.3. Hardships and Challenges

Several challenges were encountered during the project:

- **Topology Scaling in OMNeT++**: Creating and managing large-scale topologies (especially the mega-scale variant) required significant attention to NED scripting and parameter management. Debugging connectivity issues between modules was time-consuming.

- **Buffer Configuration**: The default modules did not allow for dynamic buffer adaptation. Custom modifications were needed to explore queueing behavior thoroughly, particularly under bursty traffic.

- **Result Variability**: Due to the probabilistic nature of traffic and MAC learning, repeated simulations sometimes produced variable results. Multiple runs were essential to produce statistically meaningful trends.
- **INET Code Complexity**: Navigating INET's source code to locate module control logic was challenging and time-intensive.

4.4. Assumptions

Several assumptions were made to constrain the scope and complexity of the project:

- **Uniform Traffic Generation**: All hosts were assumed to generate traffic at a constant rate to random destinations, without considering traffic locality or temporal bursts.

- **FIFO Queuing**: All switches used simple FIFO queueing mechanisms without Active Queue Management, which may not reflect advanced data center switching behavior.

- **Static Topologies**: Network topology and host placement were fixed during each simulation run, with no dynamic changes or failures modeled.

- **Ideal Links**: All links were assumed to be error-free, with no jitter or loss due to physical layer issues.

- **Simplified ECMP**: Equal-Cost Multi-Path routing was implemented as random spine selection, which approximates but does not fully emulate real-world ECMP hash-based forwarding.

5. Conclusion

This study successfully leveraged OMNeT++ to model and evaluate a Leaf-Spine data center network under a variety of load and buffer conditions. Through systematic simulations, the architecture's strengths and limitations were explored in depth.

Key conclusions drawn from the results include:

- **Predictable Scalability:** The Leaf-Spine architecture consistently demonstrated low-latency communication and effective load balancing across all tested scales. Its symmetrical design and ECMP routing allowed the network to handle increased demand with minimal congestion particularly in large and mega scale topologies.

- **Critical Role of Buffer Management:** Packet loss was heavily influenced by buffer sizes, particularly at Leaf switches where traffic converges. Adequate buffer provisioning is essential to prevent performance degradation under high load conditions.

- **Load-Driven Behavior:** While the architecture scaled well, performance began to deteriorate rapidly under extreme load when buffers were undersized. This illustrates the importance of aligning hardware resources with traffic demands, and scale up the number of spines to avoid bottleneck.

- **Simulation as a Design Tool:** OMNeT++ proved to be an effective platform for modeling complex network behaviors, offering the flexibility to explore detailed performance metrics like Flow Completion Time and queue dynamics.

Overall, the findings reinforce the Leaf-Spine topology as a robust and scalable backbone for modern data centers, with clear design trade-offs that can be tuned based on workload and resource constraints.