CRC Optimization for Error Detection in High-Speed Networks

Anurag Wagh, 231CS166
Praveen Yadav, 231CS243
Aaditya Yadav, 231CS102

February 12, 2025

1 Overview

Cyclic Redundancy Check (CRC) is a widely used error detection mechanism in data link layer protocols such as Ethernet and Wi-Fi. In high-speed networks, optimizing CRC computations is crucial for maintaining efficiency and reducing latency. This project focuses on implementing and optimizing CRC algorithms for error detection in high-speed networks. By using open-source tools such as Python and C for implementation, NS3 for simulation, and Wireshark for packet analysis, we aim to analyze the performance of CRC under different network conditions. The goal of this project is to compare standard CRC implementations with optimized versions using lookup tables (LUTs) and hardware-efficient techniques. We will measure the performance in terms of execution time, error detection accuracy, and network overhead in simulated environments.

2 Open Source Tools

- **Python** - Used to implement a basic CRC-16 algorithm for error detection.
- **C** - Used to optimize CRC computations for high-speed processing.
- **NS3** - Used to simulate network environments and introduce error conditions.
- **Wireshark** - Used to analyze network packets and verify CRC integrity in real-time.

3 References

- "High-Speed CRC Computation for Network Error Detection," IEEE Transactions on Communications, 2019.
- "Efficient CRC Algorithms for Real-Time Networking," ACM Networking Journal, 2020.