

**Initial Project Report**

**ES-221**

**Group Members**

2023534 **– Umair Afridi**

2023572 **– Neshmiya Farooq**

2023528**- Taha Farooq**

### ****Note: This initial project is open to changes as it is in initial stage****

### ****Network Packet Simulator and Routing System****

#### ****Introduction****

In modern networking, ensuring data packets reach their intended destinations with minimal latency and packet loss is critical. Efficient routing algorithms like Dijkstra's and simulation of packet loss and retransmission are key elements in network management. This report discusses the design and implementation of a network packet simulator using data structures such as graphs, priority queues, stacks, queues, and linked lists. The system aims to simulate packet transmission, packet loss, and the shortest path routing for data packets.

#### ****Problem Statement****

The key challenge in networking is dynamically determining the optimal path for packet transmission while adjusting to various network conditions such as congestion, latency, and link failures. This simulation addresses these challenges by employing well-known algorithms like Dijkstra's and simulating packet loss, retransmission, and sorting mechanisms to maintain optimal performance.

#### ****System Design and Data Structures****

The system leverages several fundamental data structures to manage network topology, packet transmission, and routing. The primary data structures used in the simulation include:

1. **Graph (Adjacency List Representation)**:
   * The network is modeled as a graph where each node represents a router, and the edges represent the network links between routers. Each edge is weighted based on latency, and the graph allows for the dynamic addition of edges to simulate real-world networks.
2. **Priority Queue (Min-Heap)**:
   * This structure is used in Dijkstra’s algorithm to select the next router with the shortest path. The priority queue allows for efficient selection of the router with the minimum distance, which is essential for the performance of the algorithm.
3. **Queue (FIFO Structure)**:
   * The queue is used to simulate packet buffering and transmission order. The first packet that enters the queue is also the first to be processed, following the FIFO (First In, First Out) principle.
4. **Linked List**:
   * A linked list is utilized to manage packet retransmissions. In case of packet loss, the packets are stored in the list for retransmission, simulating how packets are managed in real-world networking scenarios.
5. **Stack (LIFO Structure)**:
   * A stack is used to simulate LIFO (Last In, First Out) processing of packets. This structure is used to manage situations where the order of processing packets needs to be reversed.
6. **Sorting Algorithms (Selection Sort and Bubble Sort)**:
   * These algorithms are used to sort router distances to showcase how routing decisions can be based on the ordered list of distances.

#### ****Algorithmic Approach****

The system implements two key algorithms to simulate network routing:

1. **Dijkstra’s Algorithm**:
   * Dijkstra’s algorithm is used to compute the shortest path between routers based on link latency. It uses a priority queue (min-heap) to select the router with the shortest path efficiently. The algorithm calculates the minimum distance to all other routers from a given source router and outputs the shortest path to each.
2. **Packet Loss Simulation & Retransmission**:
   * Packet loss is simulated with a random function where there is a certain probability (e.g., 10%) that a packet will be lost during transmission. If a packet is lost, it is added to a retransmission list and reprocessed, ensuring that the system can recover from packet loss and maintain reliable communication.

#### ****Key Functions of the Simulation****

* **Graph Creation**:
  + The graph is initialized by creating nodes (representing routers) and connecting them with weighted edges (representing network links with associated latency).
* **Shortest Path Calculation**:
  + The system computes the shortest path from a specified source router to all other routers using Dijkstra’s algorithm. This is useful for understanding how data packets can be routed efficiently through the network.
* **Packet Transmission**:
  + Packets are transmitted through the network, and their status (successfully transmitted or lost) is determined using the packet loss probability. Lost packets are added to a retransmission list for reprocessing.
* **Queue and Stack Management**:
  + The system simulates packet buffering using a queue and packet processing using a stack. The queue follows the FIFO order, while the stack processes packets in reverse order (LIFO).
* **Sorting of Router Distances**:
  + The system demonstrates the application of sorting algorithms like selection sort and bubble sort on router distances calculated by Dijkstra’s algorithm. This allows the simulation to order routers based on their distances from the source router.

#### ****Challenges Addressed by the Simulation****

1. **Routing Optimization**:
   * By implementing Dijkstra’s algorithm, the simulation helps identify the most efficient paths for packet transmission, reducing latency and improving overall network performance.
2. **Packet Loss Handling**:
   * Simulating packet loss and retransmission highlights the importance of reliability in network communication. The system demonstrates how lost packets can be handled and retransmitted without disrupting the network's operation.
3. **Data Structure Implementation**:
   * The use of different data structures, such as priority queues, queues, stacks, and linked lists, demonstrates how various abstract data types can be applied in practical network simulations. Each structure is used in a specific context to manage packets and network routing efficiently.
4. **Algorithm Efficiency**:
   * Sorting algorithms like selection sort and bubble sort are employed to showcase how routing decisions can be optimized further by sorting router distances. This helps in understanding the practical application of algorithms in routing systems.

#### ****Simulation Results****

The simulation successfully computes the shortest paths between routers using Dijkstra’s algorithm. It also simulates packet loss with a given probability and retransmits lost packets. The system outputs the transmission status of each packet, displaying whether it was transmitted successfully or lost. Additionally, the system shows the sorted list of router distances, demonstrating how network routing decisions can be optimized.

Expected Output:

Shortest Paths from Router 0 (Dijkstra's):

To Router 0: Distance = 0

To Router 1: Distance = 7

To Router 2: Distance = 9

...

Packet Queue: Packet ID: 1 Packet ID: 2 Packet ID: 3

Packet Stack (LIFO): Packet ID: 5 Packet ID: 4

Packet ID: 1 lost. Retransmitting...

Packet ID: 2 transmitted successfully.

...