Int. J. Advanced Networking and Applications
Volume: 16 Issue: 06 Pages: 6647-6658 (2025) ISSN: 0975-0290

AI-Enhanced QoS Routing in Software Defined Networks for Smart IoT Applications

Mr. R. Suresh[1*]
[1*] Department of Computer Science, Nandha Arts and Science College (Autonomous), Erode, Tamil Nadu.
Email: sureshrctech16@gmail.com
Dr. P. Nirmaladevi[2]
[2] Department of Computer Applications, Nandha Arts and Science College (Autonomous), Erode, TamilNadu,
Email : ndevi71@gmail.com

ABSTRACT

A QoS-based routing algorithm provides routing solutions that ensure quality requirements by leveraging knowledge of network resource availability to establish optimal paths based on multiple metrics, thereby delivering adequate quality of service (QoS) for significant application flows. Additionally, it continuously monitors and adapts to fluctuations in QoS parameters across network links. One of the most effective approaches for managing large-scale and complex networks is Software Defined Networking (SDN), which simplifies network policy implementation by centralizing control at the network's top level rather than embedding policies in individual network devices. The SDN architecture consists of two vertically connected segments: the data and control planes in the position of providing data, which manages policies and control mechanisms. In scalable and dynamic network contexts, SDN-based QoS routing significantly improves upon traditional standards in terms of energy usage, sender waiting times, network lifetime, routing effectiveness, and duplicate packet handling.

Keywords – **Artificial Intelligence (AI), Dynamic Network, Internet of Things (IoT), Quality-of-service (QoS), Software-defined networking (SDN).**

Date of Submission: February 24, 2025 Date of Acceptance: April 22,2025

I. INTRODUCTION

The rapid expansion of Internet of Things (IoT) devices has created until unheard-of need for dependable, scalable, and effective network connection [1,2]. Quality of service (QoS) is vital in smart IoT applications as they usually entail real-time data transmission, important decision-making procedures, and energy-sensitive activities [3,4]. Maintaining constant network performance in such environment requires advanced routing systems that can adjust to dynamic situations and guarantee beneficial service quality as seen in Figure 1.

By separating the control and data planes, software defined networking (SDN) has become a transforming method of network administration [5,6]. Because of centralized policy administration and dynamic reconfiguration of network resources made possible by this separation, SDN especially fits for advanced, large-scale IoT networks [7]. By providing a worldwide perspective of the network, SDN assists more intelligent routing choices that can better handle the particular QoS needs of smart IoT applications that expressed in the IoT network architecture based on SDN [8,9].

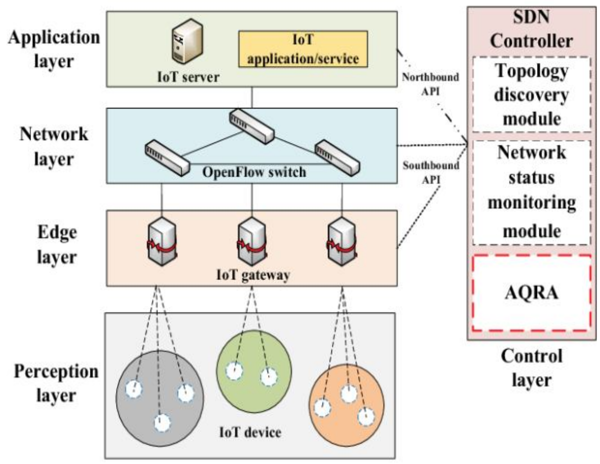

Figure 1. SDN-based IoT network architecture

Through enabling the network to learn from past data and forecast future network situations, the combination of artificial intelligence (AI) with SDN improves routing capacities even more [10–12]. Based on real-time insights, AI-driven algorithms may dynamically reroute channels, proactively change to fit changes in network performance, and automate decision-making procedures [13-15]. This combination of artificial intelligence with SDN not only increases routing efficiency but also general network dependability and responsiveness, which are crucial for controlling the many and often shifting needs of IoT settings [16].

The integrating AI with SDN can enhance routing decisions by enabling real-time network adaptations. However, beyond this theoretical mention, there is no detailed explanation, framework, or implementation of any AI algorithm for QoS routing in the main sections. The manuscript lacks practical examples, models, or performance evaluations related to AI-driven QoS mechanisms. As a result, the core focus remains on SDN and routing classifications, not on the integration of AI in QoS routing.

This paper's innovation is constrained, since it mainly compiles and categorises pre-existing information regarding Software Defined Networking (SDN), diverse routing algorithms, and Quality of Service (QoS) principles, without introducing any innovative model, technique, or framework. Despite the title referencing AI-enhanced QoS routing, the article predominantly elaborates on fundamental SDN architecture, various routing algorithms, and QoS parameters via an extensive literature review and theoretical discourse. The paper lacks new simulation results, experimental validations, or innovative AI-based routing approaches, rendering it more of a compilation and review than a presentation of a novel technological contribution.

The key contributions of this paper are as follows: (1) it provides a comprehensive overview of Software Defined Networking (SDN) architecture, including its layers, APIs, deployment models, and advantages; (2) it classifies and explains various routing algorithms—adaptive, non-adaptive, and hybrid—used in network communication; and (3) it highlights essential Quality of Service (QoS) parameters and their significance in modern networking, particularly in IoT-based applications.

This paper examines the idea of AI-enhanced QoS Routing within Software Defined Networks, especially designed for smart IoT devices. The suggested method seeks to maximize network performance, lower latency, and improve resource use by using the adaptive and predictive features of artificial intelligence. Through addressing the inherent difficulties with operating advanced IoT networks, the combination of these innovative technologies promises to open the path for more resilient and effective smart infrastructures.

II. LITERATURE REVIEW

Xuwei Yang (2021) [17] produced the Rounding-Based Algorithm for PRLF (RDBP) to get high dependability and throughput in SDN. Large-scale network simulations showed that the suggested method decreased the maximum number of needed flow entries by 53.1% compared to current methods and increased network performance by over 48%. The study guaranteed forwarding dependability by optimizing link loads, therefore addressing the dependable flow routing issue. Reducing the maximum link load assists the algorithm to improve network resilience to failures and maintains SDN environments more dependable and efficient.

Maruthupandi J (2021) [18] suggested the Distinct Network Yarning (DISNEY) Routing Protocol to improve routing efficiency in Mobile Ad-Hoc Networks (MANETs) under control of Software-Defined Networking (SDN). Using data transmission rate optimization, throughput, and packet transmission delay, the method emphasizes route manipulation awareness to enhance data transmission. Experimental statistics show a packet transmission latency of 0.65% at node 12s, a throughput of 95% at node 18s, and a data transmission rate of 86% at node 12s. The research highlighted that a feasible option for MANET applications because the combination of SDN's control and data plane designs greatly enhances routing and data transfer.

Majda Omer Elbasheer (2021) [19] presented a QoS-based routing method for SDN, the Two Lowest Loss-Widest Paths (TwoLLWPs) algorithm. The research examined packet loss rates (PLR) over many pathways and assessed SDN network connection capacity at 100 Mb/s to determine that while other links had a loss rate between 0% and 2%, loss was absent on the longest paths. Analysis of the impacts on HD and SD videos, we found that the structural similarity index (SSIM) declined dramatically when PLR approached 0.05%. Emphasizing the importance of strong bandwidth and loss-aware routing techniques in SDN-based networks, the research revealed that HD videos were more sensitive to quality deterioration owing to packet loss.

Abeer A. Z. Ibrahim (2022) [20] introduced in SDN Multi-Control Architectures a Multi-Objective Routing Mechanism for Energy Management Optimization. Three algorithms: multi-level, energy-aware routing, and mapping, routing selection algorithm, and shortest path routing selection were used in the method. Up to 65% more energy was saved using the approach compared with conventional CDBEAR and DBEAR techniques. As traffic and nodes grew, the Energy-Aware Routing Multi-Level Protocol (EARMLP) also attained up to 70% energy savings. The research concluded that improving network energy savings while preserving resilience and efficiency depends critically on the position of SDN's controller and load balancing.

Van Tong and Sami Souihi (2022) [21] developed for large-scale networks an SDN-Based Application-Aware Segment Routing Mechanism. By 64.39% over traditional benchmarks, the segment routing (SR) technique greatly lowered overhead. This system guaranteed rigorous adherence to service level agreements (SLA) by allowing alternative routing principles depending on application criteria. The study emphasized the capacity of segment routing to maximize network efficiency and QoE while lowering network congestion. Future research will concentrate on putting a root cause analysis system into use to improve routing rules even further.

Xu Li (2022) [22] introduced the Joint Routing and Task Placement (JRTP) algorithm to get cost-minimized transmission in SDN-based Space-Terrestrial Integrated

Volume: 16 Issue: 06 Pages: 6647-6658 (2025) ISSN: 0975-0290

Networks (STINs.). In single-source cases, the algorithm decreased bandwidth utilization by 53.5% and transmission costs by up to 66% by means of more effectively task allocation techniques and traffic rate reductions. The method reduced network congestion and raised general efficiency by carefully distributing tasks across network nodes. The research underlined the need of traffic optimization for rapidly expanding space-terrestrial applications.

Daniela M. Casas-Velasco (2022) [23] suggested DRSIR based on Deep Reinforcement Learning-based routing for SDN using Deep Q-learning. Reducing mean connection latency by up to 73% and throughput losses by up to 51%, the method far exceeded conventional routing techniques. D RSIR attained a 17% lower stretch value than variants of Dijkstra's algorithm, therefore demonstrating its effectiveness in dynamic routing. The efficiency of the DRSIR and RSIR agents was further shown by their use of just 29%-32% of CPU capacity and 279-287 MB of storage memory. Future studies intend to expand the model for multi-path routing and scalability enhancements for large-scale networks.

Xin Yang (2023) [24] produced three algorithms: Routing with IACO, Traffic Request Order Optimization with ISA, and SL and SB Assignment under Link Load Balancing for SD-SCNs. With 490 Tb/s in basic mesh networks and up to 650 Tb/s in NSF networks, the model showed exceptional throughput performance. The research highlighted how well SDN might maximize device use and control infrastructure expenses. The study highlighted the need of addressing RMSSA issues in large-scale networks to improve scalability and efficiency.

Muhammad Umar Farooq (2023) [25] set out the SDORP (SDN-Based Opportunistic Routing for Asynchronous Wireless Sensor Networks). To improve routing performance, the paper presented flow instantiation and network starting techniques. Duplicate packet management, energy usage, sender waiting times, routing efficiency, and the suggested strategy significantly increased the network's resilience. The work shown how flexible management solutions in wireless sensor networks (WSNs) might be given by SDN while preserving dependable connectivity.

Rohit Kumar Das (2023) [26] presented the Edge Controller-Assisted SDN Architecture using the ESDoT Flow Management Algorithm for IoT. Architectural performance was assessed against models of random computational route selection (RSCM) and standard SDN (CSDN). Results revealed an 18% give in round-trip time (RTT), an 82% drop in packet loss, and a 20% drop in end-to- end latency. System throughput also dropped 32%. The paper suggested ways to deal with heterogeneous traffic and underlined the benefits of using edge-enabled SDN controllers to maximize IoT data speed.

Martin Slabber (2024) presented a Multi-cast Traffic Engineering Algorithm based on SDN for Radio Telescope Data Networks. The research modeled complete sub-array use by increasing the number of receptors from 64 to 90, therefore evaluating network scalability. Results revealed that telescope capabilities were much improved by SDN's monitoring and management of network performance. To maximize network performance even further, the research suggested considering switch and port buffer use in bandwidth allocation.

Feiyan Li (2024) [27] developed the Q-Learning-Assisted Trust Routing Scheme for SDN-Based Underwater Acoustic Sensor Networks. Based on packet delivery ratios (PDR), the method found forwarding sets that dynamically modified routing pathways thereby preserving network performance.

P. K. Udayaprasad (2024) [28] integrated Genetic Algorithm (GA)-based route deployment, PSO-based clustering, and ABC-based optimal path selection using Distributed SDN-AI for Large-Scale IIoT Networks. According to the research, energy use dropped by 78%, network lifespan rose by 63%, and high-volume data transmission efficiency climbed by 95%. The study underlined how SDN assists in maximizing big-scale industrial IoT (IIoT) networks and recommended further study of SDN controllers such as Floodlight and NOX for more effectively coordination and efficiency.

E. Shahr (2024) [29] using many worst-case response time (WCRT) calculation techniques, suggested a Scalable Real-Time SDN-Based MQTT Framework for Industrial Applications. While TA obtained tighter scheduling boundaries at the expense of more complexity, results showed that the heuristic approach (HA) attained execution times 26.5% lower than the standard technique (TA). With future work aimed on balancing scalability and computational complexity, the research underlined the ability of the framework to maximize network performance and resource economy.

Emphasizing enhanced network efficiency, dependability, energy optimization, and scalability across many applications, including MANETs, IoT, industrial IoT, underwater networks, and space-terrestrial networks, the reviewed research show the developments in QoS-based routing for SDN. Various AI-driven and reinforcement learning models such as Deep Q-Learning, Genetic Algorithms, PSO, and Opportunistic Routing have been integrated to enhance routing decisions, resource allocation, and network adaptability. The studies demonstrate significant improvements in packet transmission delay, throughput, network lifetime, energy consumption, and link load balancing, proving SDN’s capability to optimize network management, reduce infrastructure costs, and enhance data flow control. Future research focuses on multi-path routing, large-scale network expansion, improved AI-driven automation, and enhanced security mechanisms to further refine SDN-based routing strategies.

III. SOFTWARE-DEFINED NETWORKING

SDN is an innovative method of computer network management. Hardware components like switches and routers, which may be complicated and difficult to maintain, are traditionally used for network management. SDN revolutionizes this process by separating network control (decision-making about data flow) from data movement itself. As a networking architectural method, SDN enables software-based administration and control, allowing applications to program the entire network's behavior through open APIs in a centralized manner. With its dynamic and programmatically efficient design, SDN enhances network performance and monitoring, streamlining administration processes. This is achieved by dividing the control plane's data plane, which determines traffic flow, which handles packet transmission to designated destinations.

1. Data Plane

The data plane manages all operations related to data packets sent by end users. Its key functions include:

- Packet forwarding to ensure data reaches the correct destination.
- Breaking apart and reassembling data for efficient transmission.
- Packet replication to enable multicasting when needed.

2. Control Plane

The responsibility of the control plane is all tasks that support data plane operations but do not directly involve user data packets. It essentially serves as the network's brain and is responsible for:

- Creating routing tables to optimize network paths.
- Defining policies for packet handling to enhance traffic management.

Key Advantages of SDN

- Enhanced Network Connectivity: Faster data interchange and seamless integration let SDN significantly improve internal communication, sales, and service operations.
- Accelerated Application Deployment: It ensures effective scalability by enabling new applications, services, and business models to be quickly implemented.
- Stronger Security: By allowing separation of devices depending on security criteria and improving network visibility, SDN gives operators greater flexibility and control.
- Better Speed and Control: Rapidly speed and efficiency are provided by SDN using a software-based controller that is open and standard than by conventional networking setups.

Modern network management finds uses for SDN as this change in network design guarantees flexibility, scalability, and improved control.

3.1. SDN Architecture

Control plane and data plane separation, SDN is an advanced network design that offers centralized network administration and more flexibility [30]. Unlike conventional networks where network equipment like routers and switches perform both control and forwarding tasks, SDN presents a software-based controller that dynamically configures data pathways and controls network traffic. In network operations shown in Figure 2, this separation allows programmability, automation, and scalability.

SDN architecture consists of three primary layers:

- Application Layer (User Applications and Services)
- Control Layer (SDN Controller)
- Infrastructure Layer (Network Devices)
- For communication between these levels, SDN also depends on northbound and southbound APIs.

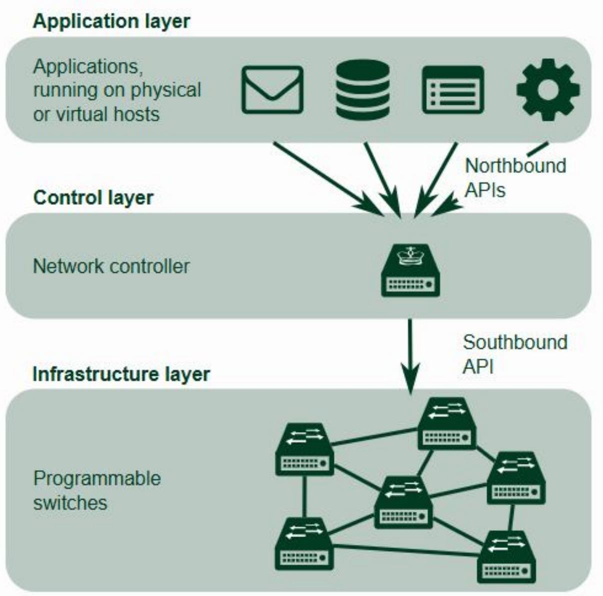

Figure 2. SDN Architecture

3.2. SDN Architecture Layers

1. Application Layer

Comprising user applications and network services using SDN capabilities, the Application Layer is the highest level in SDN architecture. Applications obtaining network resources, policy definition, and traffic flow optimization communicate using northbound APIs from the SDN controller.

Key Functions of the Application Layer:

- Implements network automation and orchestration
- Provides traffic engineering and security applications
- Supports network analytics and monitoring tools
- Common SDN applications include firewall policies, load balancing, intrusion detection systems (IDS), and quality of service (QoS) management.

2. Control Layer (SDN Controller)

The brain of the SDN architecture is the Control Layer also identified as the SDN Controller. It manages data forwarding decisions according on network regulations, security restrictions, and traffic situations thereby managing the whole network.

Key Functions of the Control Layer:

- Maintains a worldwide perspective of the network architecture
- Implements routing and switching policies
- Manages network-wide security enforcement
- Optimizes traffic flow and congestion control
- Includes both northbound APIs for communication with the Application Layer and southbound APIs for interaction with the Infrastructure Layer.

Popular SDN controllers include OpenDaylight, ONOS (Open Network Operating System), Ryu, NOX, and Floodlight.

3. Infrastructure Layer (Data Plane / Forwarding Layer)

Including all virtual and physical network devices, such as access points, switches, and routers, the Infrastructure Layer also identified as the Data Plane or Forwarding Plane is Based on SDN Controller instructions, these devices forward the packets according to direction.

Key Components of the Infrastructure Layer:

- SDN-enabled switches (e.g., OpenFlow switches)
- Routers and access points
- Virtual switches (e.g., Open vSwitch)

Functions of the Infrastructure Layer:

- Processes and forwards data packets
- Monitors and reports traffic statistics to the controller
- Enforces policies as defined by the SDN Controller.

3.3. SDN APIs and Protocols

1. Northbound APIs

Northbound APIs enable the Control Layer and the Application Layer to interact. These APIs let programs dynamically ask for and change network behavior.

Common Northbound APIs:

- RESTful APIs – Used for automation and integration with cloud services
- OpenStack Neutron API – Provides virtual networking in cloud environments
- Java-based APIs (e.g., ODL Controller APIs) – For building custom SDN applications

2. Southbound APIs

By connecting the Control Layer to the Infrastructure Layer, southbound APIs let the controller to guide routers and switches on data traffic handling.

Common Southbound APIs:

- OpenFlow – The most widely used protocol for controlling SDN switches
- NETCONF/YANG – Used for configuring network devices and managing network policies
- BGP-LS (Border Gateway Protocol - Link State) – Used for SDN-based routing
- P4 (Programming Protocol-Independent Packet Processors) – Allows flexible packet forwarding

3.4. SDN Deployment Models

1. Centralized SDN

In a centralized SDN paradigm, one SDN controller manages the whole network. It offers a worldwide perspective ensuring best traffic engineering, security policies, and efficient routing. Scalability might, however, be a drawback as one controller could turn into a bottleneck.

2. Distributed SDN

Multiple SDN controllers used across many network areas in a distributed SDN approach improves scalability, fault tolerance, and resilience. While coordinating with other controllers, each controller manages a subset of network devices.

3. Hybrid SDN

Combining conventional networking with SDN features, hybrid SDN lets old networks be gradually replaced. It is often used in companies when not all devices allow SDN protocols.

3.5. SDN Vs TRADITIONAL NETWORKING

In design, administration, and scalability, traditional networking and SDN differ drastically. In conventional networks, each device must be manually configured as routers and switches have the control and data planes are tightly coupled. This inflexible, hardware-dependent strategy results in complicated administration, less flexibility, and higher operating costs. Traditional networking makes interoperability and centralized management difficult as it depends on proprietary protocols from many suppliers. Furthermore, requiring human intervention are network regulations and traffic management, therefore restricting automation and adaptability in large-scale systems.

In contrast, Figure 3 demonstrates that the SDN dynamically designs network behavior by decoupling centralizing network intelligence in an SDN controller by the separation of the control and data planes. Automated traffic control, real-time network optimization, and policy enforcement were made possible by this software-driven method by use of APIs, SDN promotes programmability, thereby enhancing scalability, adaptability, and efficiency over conventional networking. In addition, SDN is an effective choice for cloud computing, data centers, and IoT-driven systems as it offers network virtualization, improved

Volume: 16 Issue: 06 Pages: 6647-6658 (2025) ISSN: 0975-0290

security, and fast implementation of services. Network agility, cost-effectiveness, and general performance are much improved in SDN by moving from hardware-based control to software-defined operations in below Table 1.

TABLE 1: SDN vs. Traditional Networking Difference

Software Defined Networking	Traditional Networking
One method of virtual networking is the Software Defined Network.	The outdated conventional networking method is known as a traditional network.
Control is centralized in a software-defined network.	Distributed control is used in traditional networks.
The network may be programmed.	This network is nonprogrammable.
The open interface is the Software Defined Network.	An interface that is closed is a classical network.
In a software-defined network, software separates the data plane from the control plane.	In a typical network, the data plane and control plane operate on the same plane.

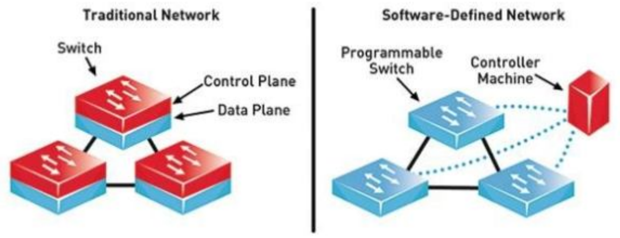

Figure 3. Traditional Networking Vs Software Defined Network

3.6. Advantages of SDN Architecture

- Centralized Network Control – Simplifies network administration and policy enforcement.
- Programmability and Automation – Allows dynamic traffic management and software-defined security policies.
- Improved Scalability – Supports large-scale networks through distributed and hybrid SDN models.
- Enhanced Security – Provides network segmentation, real-time threat detection, and flexible security policies.
- Optimized Resource Utilization – Reduces network congestion, latency, and bandwidth wastage.

3.7. Challenges in SDN Architecture

1. Controller Scalability and Latency

As networks grow, controller bottlenecks can lead to increased response times. Distributed SDN solutions aim to mitigate this issue.

2. Security Risks

As a centralized control point, the SDN controller, it becomes a prime target for cyber-attacks, such as DDoS attacks and unauthorized access.

3. Compatibility with Legacy Systems

Complete conversion can be difficult since many existing network devices do not support SDN protocols. Hybrid SDN offers a transitional solution.

4. Complexity in Network Policies

Managing policies across diverse network environments can be challenging, requiring sophisticated SDN controllers and well-defined APIs.

3.8. Future Trends in SDN Architecture

- AI and Machine Learning Integration – Enhancing SDN with intelligent traffic analysis, anomaly detection, and self-healing networks.
- Edge Computing and IoT Integration – Enabling low-latency network control for IoT and edge devices.
- Blockchain for Secure SDN – Using blockchain for decentralized authentication and data integrity.
- 5G and Beyond – SDN will play a key role in 5G network slicing, ultra-low latency communication, and dynamic service provisioning.

SDN architecture fundamentally transforms network management by providing programmability, automation, and centralized control over network resources. Through the data and control planes being separated, SDN improves scalability, security, and resource allocation while supporting emerging technologies such as IoT, edge computing, and AI-driven networking. Despite challenges in controller scalability, security, and legacy compatibility, ongoing innovations are continuously enhancing SDN’s capabilities, making it an integral part of modern network infrastructure.

IV. ROUTING ALGORITHM

When a data packet is transmitted from a source to a destination, it can take multiple possible paths. A routing algorithm mathematically determines the most efficient path, known as the "least-cost path", to direct data packets efficiently [31]. These algorithms play a crucial role in managing internet traffic, enhancing efficiency and performance network communication. Routing involves constructing a routing table, which contains information about different paths that packets can take to reach their intended destination. Different types of routing algorithms exist in the figure 4, each designed to optimize routing performance based on the network structure, traffic conditions, and application requirements. These algorithms are classified into three main categories and represented in Figure 4:

- Adaptive Algorithms
- Non-Adaptive Algorithms
- Mixed Algorithms

Volume: 16 Issue: 06 Pages: 6647-6658 (2025) ISSN: 0975-0290

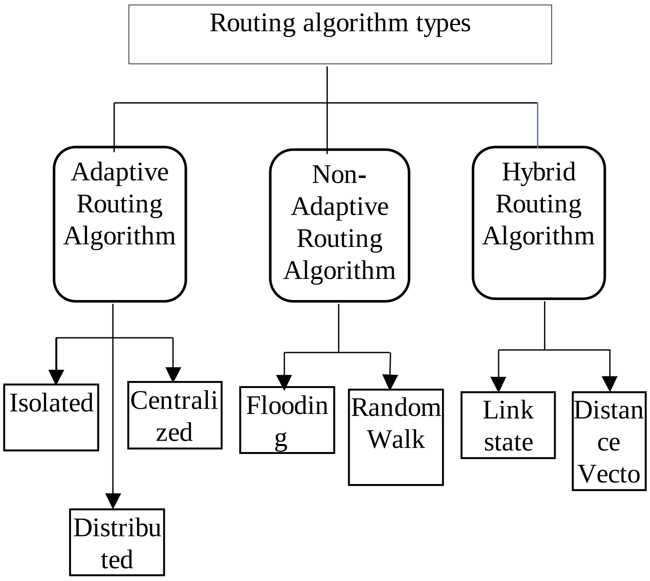

Figure 4. Types of Routing Algorithm

4.1. Types of Routing Algorithms

4.1.1. Adaptive Routing Algorithms

Adaptive routing algorithms are dynamic in nature, meaning they adjust their routing decisions according to actual network conditions, including congestion and traffic load, topology changes, or latency fluctuations [32]. These algorithms ensure that the network remains efficient by dynamically optimizing paths to avoid congestion or failures. Adaptive routing is widely used in large-scale, complex networks where constant changes occur.

Optimization Parameters for Adaptive Routing Algorithms

- Expected Transit Time – Ensuring minimal latency for packet delivery
- Number of Hops – Reducing the number of intermediary routers to reach the destination
- Distance – Choosing the shortest path to optimize network performance

Subcategories of Adaptive Routing Algorithms

1. Isolated Adaptive Routing

- Each network node makes its routing decisions independently without consulting other nodes.
- Nodes lack awareness of current network conditions, leading to possible inefficiencies.
- A drawback is that packets may get routed through highly congested paths, increasing delays.
- Examples: Backward Learning, Hot Potato Routing

2. Centralized Adaptive Routing

- All routing decisions are made at one central node; this provides a complete network perspective.
- Since all network data is stored at a single location, routing decisions are more accurate and efficient.
- However, the central node fails, the entire network is a single point of failure.
- Example: Link-State Routing Algorithm

3. Distributed Adaptive Routing

- Each node gathers network information from its neighbors before deciding the best path for packets.
- It reduces network congestion by dynamically adjusting routing paths.
- A delay can occur if there is a gap between when the node receives information and forwards packets.
- Example: Decentralized Routing Algorithm

4.1.2. Non-Adaptive Routing Algorithms

Non-adaptive routing algorithms do not alter their routing decisions after they have been made, in contrast to adaptive algorithms. These algorithms rely on predefined, static routing tables that are configured when the network is initialized. Since non-adaptive routing does not respond to changes in traffic load or network failures, it is typically used in small, simple networks where dynamic adjustments are unnecessary.

Subcategories of Non-Adaptive Routing Algorithms

1. Flooding Routing Algorithm

- Every incoming data packet is forwarded to all possible outgoing paths (except the path from which it came).
- This technique ensures that packets always reach their destination, but it increases network congestion.

Solutions to prevent packet looping:

- Hop Count Limitation – Limits the number of times a packet is forwarded.
- Spanning Tree Algorithm – Ensures packets are only forwarded along a tree-like structure.
- Sequence Numbering – Prevents duplication by marking packets with unique identifiers.

2. Random Walk Routing Algorithm

- Instead of following a fixed path, packets are randomly forwarded to one of the available neighboring nodes.
- Ensures load balancing but can lead to higher delays in reaching the destination.
- This algorithm is useful when dealing with highly dynamic or unpredictable network environments.

4.1.3. Mixed Routing Algorithms

Mixed algorithms combine elements of both adaptive and non-adaptive routing to balance efficiency and stability. The network is divided into many zones using this method, and each zone has a distinct routing algorithm.

Subcategories of Mixed Routing Algorithms

1. Link-State Routing Algorithm

- Every router creates and maintains a complete network topology map.
- The routing decisions are based on updated network information, allowing for precise and efficient routing.

2. Distance Vector Routing Algorithm

Volume: 16 Issue: 06 Pages: 6647-6658 (2025) ISSN: 0975-0290

- Every router maintains a routing table with details about all possible paths in the network.
- Routing tables are shared periodically with other routers to update path information.
- A drawback is that it may cause routing loops, leading to packet delays.
- In Adaptive and Non-Adaptive Routing Algorithms: A Key Difference

Feature	Adaptive Routing Algorithms	Non-Adaptive Routing Algorithms
Routing Decision	Adjusts dynamically based on real-time network conditions	Fixed and does not change after being established
Traffic Handling	Responds to network congestion, failures, and topology changes	Does not respond to changes in network conditions
Complexity	More complex due to real-time monitoring and decision-making	Simpler and easier to implement
Performance in Large Networks	Suitable for large, dynamic, high-data traffic networks	Suitable for small, stable networks with low data traffic
Examples	Distance Vector Routing, Link-State Routing	Flooding, Random Walk, Static Routing

Routing algorithms are essential for determining out the best path for data packets, improving network efficiency and reliability. Adaptive algorithms provide dynamic and real-time adjustments to optimize network congestion, load balancing, and performance, making them suitable for large, high-traffic environments. Non-adaptive algorithms, on the other hand, rely on predefined routing paths, ensuring simplicity and predictability but lacking flexibility to adapt to network changes.

Mixed algorithms integrate both approaches, enabling more robust and scalable network designs. Understanding these routing mechanisms is essential for modern networking, ensuring optimized data transmission, reduced latency, and efficient resource utilization.

4.2. Types of Routing Protocol in Computer Networks

To determine the optimal path for data packets to take across a network, routing protocols are essential [33]. They are designed to discover, maintain, and update routing information, ensuring efficient and reliable communication. These protocols are broadly classified into three major categories:

- Protocols for Routing: Static vs Dynamic
- Routing protocols for interior vs exterior gateways
- Link-state vs distance vector vs hybrid routing protocols

Each type of routing protocol serves a different purpose, and their selection depends on network size, topology, scalability, and performance requirements.

1. Routing information protocol (RIP)

The inner gateway protocol, or RIP, is one of the earliest protocols ever developed. Wide area networks (WANs), which are telecom networks that span broad regions, or local area networks (LANs), which are networks of connected computers in a constrained region, may both be used with it. To determine the quickest path between networks, the Routing Information Protocol (RIP) uses hop counts.

2. Interior gateway protocol (IGRP)

Cisco, a global technology business, developed IGRP. The maximum number of supported hops is raised to 100 while many of the key features of RIP are still in place. Consequently, it might function more effectively on larger networks. IGRPs stand for elegant distance-vector protocols. To function, IGRP relies on cross-indicator comparisons, which include load, reliability, and network capacity. Additionally, when things change, such as the route, its kind immediately adjusts. This assists in preventing errors that lead to an interminable cycle of data transit, known as routing loops.

3. Exterior Gateway Protocol (EGP)

In autonomous systems, the flow of data or information in several gateway hosts is assisted using external gateway protocols (EGP). Specifically, it's used to provide routers with the room they need to transfer data across domains, like the internet.

4. Enhanced interior gateway routing protocol (EIGRP)

This kind falls under the categories of inner gateway, distance vector routing, and classless protocols. To maximize effectiveness, it uses the reliable transport protocol and the diffusing update technique. A router can gather data and save it for later use by using the tables of other routers. When something changes, each router notifies its neighbor so that everyone is aware of the active data pathways. It prevents miscommunication between routers. Border Gateway Protocol (BGP) is the exclusive external gateway protocol.

5. Open shortest path first (OSPF)

The shortest path first (SPF) algorithm is used by OSPF, an inner gateway, link state, and classless protocol, to ensure efficient data transmission. It maintains the most recent versions of many databases that provide network-wide data and topology tables. The advertising gives thorough data of the path's length and potential requirements for resources, which seem like studies. OSPF updates routes in response to topology changes using the Dijkstra algorithm. It additionally employs authentication processes to protect its data from modifications or network assaults. OSPF's scalability features make it potentially beneficial for network enterprises of all sizes.

6. Border gateway protocol (BGP)

The initial objective was to replace EGP with a different outer gateway protocol named BGP. Since it uses the optimum route selection approach to executes data package transfers, it is also a distance vector protocol. BGP describes Internet-based communication. A large network of connected, independent systems that composed the internet. By registration with the Internet Assigned Numbers Authority, each autonomous system is assigned an autonomous system number (ASN).

4.3. Overview: Difference Between Routing and Flooding

Two separate techniques used in network communication to distribute data packets are routing and flooding; each has unique procedures, benefits, and restrictions.

1. Routing

Data packets travel a preset or dynamically selected route to reach their destination via the selective and efficient process known as routing. Based on network topology, traffic load, and routing parameters, routing algorithms distance vector, link-state, hybrid protocols determine the optimal or least-cost path: Routing avoids unnecessary transmissions, therefore guaranteeing effective use of network resources.

Key Characteristics of Routing:

- Path Optimization**:** Chooses the most efficient route based on metrics like hop count, bandwidth, and latency.
- Controlled Packet Transmission**:** Packets are forwarded only along specific routes, preventing unnecessary congestion.
- Scalability: Works well in both small and large networks with structured communication paths.
- **Examples:** RIP, OSPF, EIGRP, BGP.

2. Flooding

Every data packet in a non-selective flooding system is delivered to all potential outgoing connections, therefore guaranteeing that it reaches every node in the network. Although flooding is extremely redundant and assures packet delivery even in defective or fast changing networks because it is not depending on routing tables.

Key Characteristics of Flooding:

- Broadcast-Based: Every incoming packet is forwarded to all neighboring nodes, except the one it arrived from.
- Ensures Packet Delivery: Used in network discovery and highly dynamic environments where routing tables are unreliable.
- High Redundancy & Congestion: Causes packet duplication, leading to increased bandwidth usage and potential network overload.
- Examples: Used in ARP request broadcasts, network discovery, and some emergency communication scenarios.

TABLE 2: Key Differences Between Routing and Flooding

Feature	Routing	Flooding
Method	Selective path selection based on routing protocols	Broadcasts packets to all available paths
Efficiency	Optimized for minimal delays and bandwidth usage	Redundant transmissions cause high bandwidth consumption
Scalability	Scalable for large networks with controlled traffic	Becomes inefficient as network size increases
Use Case	Used in structured networks for efficient communication	Used in emergency broadcasts and network discovery
Packet Duplication	No or minimal duplication	High packet duplication, leading to congestion
Example Protocols	OSPF, BGP, RIP, EIGRP	ARP Request, Network Discovery

Routing is a targeted and efficient method for directing data packets along the best path, making it ideal for stable, structured networks in above Table 2. Flooding, on the other hand, is an exhaustive and non-optimized approach that ensures maximum coverage but at the cost of excessive bandwidth usage and network congestion. While routing is used in almost all communication networks, flooding is generally reserved for special cases like network discovery, emergency broadcasts, and failure recovery scenarios.

3. Quality-of-service (QoS):

Strategies for traffic management that provide predictable or guaranteed performance for applications, sessions, or traffic aggregates, or that differentiate performance according to the requirements of the application or network operator, are discussed [34]. Packet delay and various kinds of losses are the core ideas of Quality of Service (QoS).

QoS Specification: Throughput, Error Rate, Variation, and Delay

3.1. Types of Quality of Service

- Stateless Solutions: Routers have the advantage of being reliable and scalable even though they do not maintain an exact log of all traffic. However, its services are insufficient as there is no guarantee as to what kind of performance or delay we will encounter in a particular application.
- Stateful Solutions: Routers maintain a per-flow state, which renders them significantly less resilient and scalable, because flow is essential for providing Quality-of-Service, or robust services like assured

services and high resource utilization, as well as protection.

QoS Parameters

- Packet loss: Routers and switches begin losing packets as network connections get congested.
- Jitter: Route modifications, network congestion, and time drift are the causes of this. The quality of voice and visual communication may be impacted by excessive jitter.
- Latency: The time it takes for a packet to travel between locations. The latency should be as near zero as is feasible.
- Bandwidth: The ability of a network communications connection for transmitting the most of data when a certain period of time.
- Mean Opinion Score: This measure of voice quality uses a five-point rating system, with five being the highest quality.

- How Does Quality of Service Operate?
  - Quality of Service (QoS) guarantees that key applications run smoothly even with constrained network resources.
  - Packet Marking: To distinguish between different service types, QoS stamps packets. It can differentiate between audio, video, and data transmission, for instance.
  - Virtual Queues: Based on priority, routers establish unique virtual queues for every application. Apps that are essential are allotted bandwidth.
  - Handling Allocation: Quality of Service determines the packet processing order, guaranteeing that each application has the right amount of bandwidth.

3.2. Benefits of Quality of Service (QoS)

Through providing critical traffic top priority, lowering latency, and efficiently managing bandwidth, Quality of Service (QoS) is a network management tool that guarantees best performance [35]. To maintain a constant and high-performance user experience, it is extensively utilized in real-time applications like VoIP, video streaming, online gaming, and cloud computing.

1. Improved Network Performance

- **Prioritization of Traffic:** Ensures that vital applications (like video conferencing and VoIP) have low latency and the necessary bandwidth.
- Minimized Packet Loss: Prevents data loss by allocating resources efficiently.
- Optimized Bandwidth Usage: Distributes network resources based on demand, avoiding congestion.

2. Enhanced User Experience

- Smooth Video Streaming & VoIP Calls: Eliminates lag, jitter, and dropped connections, ensuring high-quality communication.
- Faster Application Response Times: Reduces latency in cloud applications and online services.
- Stable Online Gaming & Remote Work: Guarantees a lag-free experience, essential for real-time interactions.

3. Increased Reliability & Stability

- Consistent Performance: Provides predictable network behavior for mission-critical applications. Network Traffic Control: Prevents network congestion by allocating resources dynamically. Resilience to Network Failures: QoS can reroute high-priority traffic through alternative paths if a failure occurs.

4. Better Business Efficiency

- Improved Productivity: Employees can access cloud applications, video conferencing, and collaboration tools without network delays.
- Optimized Resource Utilization: Businesses can allocate network resources to high-value tasks, preventing unnecessary bandwidth consumption.
- Cost Savings: Reduces the need for expensive bandwidth upgrades by making efficient use of existing resources.

5. Security and Compliance Benefits

- Controlled Data Flow: Prevents malicious traffic or cyberattacks from affecting critical services.
- Guaranteed Service Levels: Meets Service Level Agreements (SLAs) in enterprise networks.
- Regulatory Compliance: Ensures organizations meet network security and quality standards for industries like healthcare, finance, and telecom.

QoS plays a vital role in optimizing network performance, enhancing user experience, and ensuring business continuity. By managing bandwidth allocation, traffic prioritization, and minimizing delays, QoS helps maintain a stable and efficient network, making it essential for modern cloud services, enterprise operations, and multimedia applications.

V. Conclusion

AI-enhanced QoS routing in Software-Defined Networks (SDN) provides an intelligent, adaptive, and efficient approach to managing dynamic network conditions in smart IoT applications. By leveraging AI-driven decision-making, SDN optimizes traffic routing, resource allocation, and latency reduction, ensuring high-performance network operations. Real-time congestion modifications are made possible by the use of machine learning techniques, bandwidth fluctuations, and security threats, enhancing overall network reliability and scalability. Future enhancements in this field will focus on deep reinforcement learning for self-optimizing routing, blockchain-based security for tamper-proof QoS management, and edge AI to process routing decisions closer to IoT devices, reducing latency and computational overhead. Additionally, integrating 6G networks and quantum computing into AI-driven SDN frameworks will further elevate network efficiency, automation, and adaptability in evolving IoT ecosystems.

ACKNOWLEDGEMENT

AUTHOR CONTRIBUTION

Mr. R. Suresh: Methodology, Writing-Original Draft, Conceptualization
Dr. P. NirmalaDevi: Data Curation, Supervision, Validation

FUNDING

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

CONFLICT OF INTEREST

The authors declare that they have no conflicts of interest related to this research.

REFERENCES

[1] M. Slabber, N. Ventura, and J. Mwangama, SDN-based multicast traffic engineering for radio telescope data networks. *IEEE Networking Letters*, *6*(1), 2023, 65-69.

[2] M. Rostami, and S. Goli-Bidgoli, An overview of QoS-aware load balancing techniques in SDN-based IoT networks. *Journal of cloud computing*, *13*(1), 2024. 89.

[3] M. F. Osman, M. R. M. Isa, M. A. Khairuddin, M. A. M. Shukran, and N. A. M. Razali, A Novel Network Optimization Framework Based on Software-Defined Networking (SDN) and Deep Learning (DL) Approach. *JOIV: International Journal on Informatics Visualization*, *8*(4), 2024. 2082-2089.

[4] G. C. Deng, and K. Wang, An application-aware QoS routing algorithm for SDN-based IoT networking. In *2018 IEEE Symposium on Computers and Communications (ISCC)*, 2018, 00186-00191.

[5] B. Isyaku, K. bin Abu Bakar, N. M. Yusuf, M. Abaker, A. Abdelmaboud, and W. Nagmeldin, Software defined wireless sensor load balancing routing for internet of things applications: Review of approaches. *Heliyon*, 2024.

[6] M. A. Zormati, H. Lakhlef, and S. Ouni, Routing Optimization Based on Distributed Intelligent Network Softwarization for the Internet of Things. In *Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing*, Catania, Ital, 2024, 1757-1764.

[7] F. Paolucci, A. Sgambelluri, F. Cugini, and P. Castoldi, Network telemetry streaming services in SDN-based disaggregated optical networks. *Journal of Lightwave Technology*, *36*(15), 2018, 3142-3149.

[8] T. P. Carvalho, F. A. Soares, R. Vita, R. D. P. Francisco, J. P. Basto, and S. G. Alcalá, A systematic literature review of machine learning methods applied to predictive maintenance. *Computers & Industrial Engineering*, *137*, 2019, 106024.

[9] E. Shahri, P. Pedreiras, and L. Almeida, A scalable real-time sdn-based mqtt framework for industrial applications. *IEEE Open Journal of the Industrial Electronics Society*, 2024.

[10] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, Network Function Virtualization: Challenges and Opportunities for Innovations. *IEEE Communications Magazine*, 53(2), 2015, 90-97.

[11] B. Varghese, N. Wang, S. Barbhuiya, P. Kilpatrick, and D. S. Nikolopoulos, Challenges and Opportunities in Edge Computing. *2016 IEEE International Conference on Smart Cloud (SmartCloud)*, 2016, 20-26.

[12] S. Barré, C. Paasch, and O. Bonaventure, Multipath TCP: From Theory to Practice. *Networking 2011*, 2011, 444-457.

[13] J. Seedorf, and E. Burger, Application-Layer Traffic Optimization (ALTO) Problem Statement. *Internet Engineering Task Force*, RFC 5693, 2009.

[14] J. M. Halpern, and C. Pignataro, Service Function Chaining (SFC) Architecture. *Internet Engineering Task Force*, RFC 7665, 2015.

[15] S. Zhang, An Overview of Network Slicing for 5G. *IEEE Wireless Communications*, 26(3), 2019, 111-117.

[16] L. Velasco, S. Barzegar, F. Tabatabaeimehr, and M. Ruiz, Intent-Based Networking and Its Application to Optical Networks. *Journal of Optical Communications and Networking*, 14(1), 2022, A1-A12

[17] X. Yang, H. Xu, J. Liu, C. Qian, X. Fan, H. Huang, and H. Wang, Achieving high reliability and throughput in software defined networks. *Computer Networks*, *197*, 2021, 108271.

[18] J. Maruthupandi, S. Prasanna, P. Jayalakshmi, V. Mareeswari, and P. Sanjeevi, Route manipulation aware software-defined networks for effective routing in SDN controlled MANET by disney routing protocol. *Microprocessors and Microsystems*, *80*, 2021, 103401.

[19] M. O. Elbasheer, A. Aldegheishem, J. Lloret, and N. Alrajeh, A QoS-Based routing algorithm over software defined networks. *Journal of Network and Computer Applications*, *194*, 2021, 103215.

[20] A. A. Ibrahim, F. Hashim, A. Sali, N. K. Noordin, and S. M. Fadul, A multi-objective routing mechanism for energy management optimization in SDN multi-control architecture. *IEEE Access*, *10*, 2022, 20312-20327.

[21] V. Tong, S. Souihi, H. A. Tran, and A. Mellouk, SDN-based application-aware segment routing for

large-scale network. *IEEE Systems Journal*, *16*(3), 2021, 4401-4410.

[22] X. Li, F. Tang, Y. Zhu, L. Fu, J. Yu, L. Chen, and J. Liu, Processing-while-transmitting: Cost-minimized transmission in SDN-based STINs. *IEEE/ACM Transactions on Networking*, *30*(1), 2021, 243-256.

[23] D. M. Casas-Velasco, O. M. C. Rendon, and N. L. da Fonseca, DRSIR: A deep reinforcement learning approach for routing in software-defined networking. *IEEE Transactions on Network and Service Management*, *19*(4), 2021, 4807-4820.

[24] X. Yang, and Q. Sun, Joint RMSSA scheme and link load balancing for static SD-SCNs from model building to algorithm design. *Journal of Optical Communications and Networking*, *15*(8), 2023, 527-540.

[25] M. U. Farooq, X. Wang, A. Hawbani, L. Zhao, A. Al-Dubai, and O. Busaileh, SDORP: SDN based opportunistic routing for asynchronous wireless sensor networks. *IEEE Transactions on Mobile Computing*, *22*(8), 2022, 4912-4929.

[26] R. K. Das, N. Ahmed, A. K. Maji, and G. Saha, Edge controller-assisted SDN architecture for internet of things. *IEEE Sensors Journal*, *23*(22), 2023, 28182-28190.

[27] F. Li, G. Han, C. Lin, F. Zhang, and C. Sun, Sdn-qltr: Q-learning-assisted trust routing scheme for sdn-based underwater acoustic sensor networks. *IEEE Internet of Things Journal*, *11*(6), 2023, 10682-10694.

[28] P. K. Udayaprasad, J. Shreyas, N. N. Srinidhi, S. D. Kumar, P. Dayananda, S. S. Askar, and M. Abouhawwash, Energy efficient optimized routing technique with distributed SDN-AI to large scale I-IoT networks. *IEEE Access*, *12*, 2024, 2742-2759.

[29] E. Shahri, P. Pedreiras, and L. Almeida, A scalable real-time sdn-based mqtt framework for industrial applications. *IEEE Open Journal of the Industrial Electronics Society*, 2024.

[30] A. Hakiri, A. Gokhale, P. Berthou, D. C. Schmidt, and T. Gayraud, Software-defined networking: Challenges and research opportunities for future internet. *Computer Networks*, 75, 2014, 453-471.

[31] A. Alidadi, S. Arab, and T. Askari, A novel optimized routing algorithm for QoS traffic engineering in SDN-based mobile networks. *ICT Express*, 8(1), 2022, 130-134.

[32] R. Amin, E. Rojas, A. Aqdus, S. Ramzan, D. Casillas-Perez, and J. M. Arco, A survey on machine learning techniques for routing optimization in SDN. IEEE Access, 9, 2021, 104582-104611.

[33] G. Kirubasri, S. Sankar, D. Pandey, B. K. Pandey, V. K. Nassa, and P. Dadheech, Software-defined networking-based Ad hoc networks routing protocols. *In Software defined networking for Ad Hoc networks*, Cham: Springer International Publishing, 2022, 95-123.

[34] M. Rostami, and S. Goli-Bidgoli, An overview of QoS-aware load balancing techniques in SDN-based IoT networks. *Journal of cloud computing*, *13*(1), 2024, 89.

[35] S. M. Eswarappa, P. H. Rettore, J. Loevenich, P. Sevenich, and R. R. F. Lopes, towards adaptive qos in sdn-enabled heterogeneous tactical networks. *In 2021 International Conference on Military Communication and Information Systems (ICMCIS)*, 2021, pp. 1-8.

Biographies and Photographs

Mr. R. Suresh., is an Research Scholar, Department of Computer Science, Nandha Arts and Science College (Autonomous), Erode. He is 0ver 14 years of teaching and six years of research experience. His expertise is in Advanced Networking and Educational Technology. He is currently working as an Assistant Professor at Dr. N.G.P. Arts and Science College, Coimbatore. He also published book with ISBN and one patents.

Dr. P. Nirmaladevi, Assistant Professor, Department of Computer Applications at Nandha Arts and Science College (Autonomous), Erode. She has more than 18 years of experience as an Academician and is recognized as a renowned researcher in computer science. Over 20 research papers have been presented and published in national and international journals and conference proceedings. She also published two books with ISBN and two patents. Her research interest focuses on Network Security.