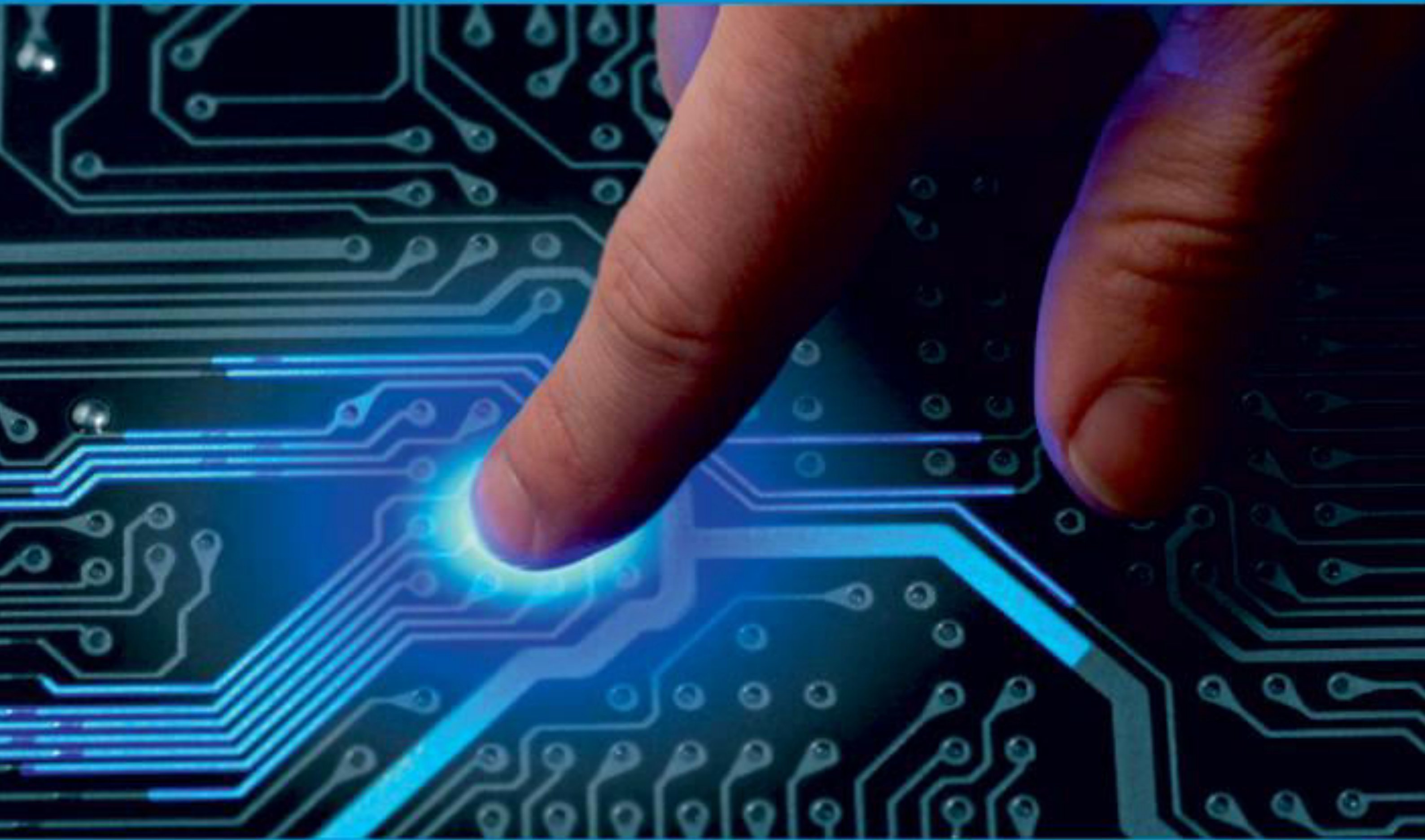
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Issue 1, January 2024

Impact Factor: 8.379

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | A Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 1, January 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1201080 |

ANN Based Traffic Prediction and Hybrid Tabu Search Optimization for High Density Route Identification in Software Defined Networks

Kranti Kumar Appari

Scholar, Department of ECE, Andhra University, Andra Pradesh, India

ABSTRACT: This paper presents an integrated approach for monitoring traffic flow in Software-Defined Network (SDN) environments and optimizing the obtained data using artificial intelligence techniques. Traditional network monitoring approaches prove inadequate for identifying traffic bottlenecks, predicting network behavior, and optimizing routing decisions in increasingly complex SDN architectures. To address these challenges, we propose a methodology combining traffic data collection using the Floodlight controller, traffic prediction using Artificial Neural Networks (ANNs), and route optimization using novel hybrid algorithms. Our experimental results demonstrate that the proposed ANN model achieves high prediction accuracy across various network topologies, with R-squared values reaching 0.97 and Mean Absolute Percentage Error (MAPE) as low as 3.1%. Furthermore, we compare four optimization algorithms—linear search, traditional tabu search, a modified tabu search, and a novel blend algorithm combining tabu search with simulated annealing—for identifying high-density traffic routes. The modified tabu search algorithm demonstrates superior performance, reducing execution time by 50% compared to linear search while maintaining 99% solution quality. The integrated system successfully identifies high-density routes with 98% accuracy and processing delays under 150 milliseconds, enabling real-time traffic management and proactive congestion prevention in SDN environments.

KEYWORDS Software-Defined Networks (SDN), Traffic Monitoring, Artificial Neural Networks, Tabu Search Algorithm, Optimization, Floodlight Controller, Machine Learning, Network Management.

I. INTRODUCTION

With the developing technology, data centers are getting bigger. Growing data centers begin to contain large volumes, complex, and disorganized information. This information in big data needs to be processed in order to be meaningful and valuable. Big data cannot be processed, managed, and stored by traditional methods. In other words, the traditional network management approach is insufficient at this stage. With a better network approach, new methods, and a wider bandwidth, this data can be processed. Software Defined Networking (SDN) is a method that meets these needs. SDN provides ease of management, hardware independence, dynamic, flexible and scalable network architecture. Therefore, this offers an effective solution to large and complex network management.

II. LITERATURE REVIEW

2.1 Evolution of Software-Defined Networks

The concept of Software-Defined Networking (SDN) emerged as a response to the increasing complexity and inflexibility of traditional network architectures. Traditional networks integrate control and data planes within the same devices, creating a tightly coupled system that is difficult to modify, upgrade, and manage (Kreutz et al., 2015). This integration has become increasingly problematic as networks grow in size and complexity, particularly in modern data centers that must accommodate diverse and dynamic workloads.

III. METHODOLOGY

3.1 Overview of the Proposed Approach

This section details the methodology employed for traffic monitoring and optimization in Software-Defined Network (SDN) environments. The proposed approach integrates data collection from network topologies, traffic prediction using Artificial Neural Networks (ANNs), and traffic route optimization using novel hybrid algorithms. Figure 2 illustrates the overall workflow of the methodology.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | A Monthly Peer Reviewed & Referred Journal |
|| Volume 12, Issue 1, January 2024 ||
| DOI: 10.15680/IJIRCCE.2024.1201080 |

The methodology consists of four main phases: (1) setting up the SDN environment and collecting traffic data, (2) developing and training ANN models for traffic prediction, (3) implementing and comparing optimization algorithms for identifying high-density traffic routes, and (4) analyzing the performance of the proposed approach.

3.2 SDN Environment Setup and Data Collection

3.2.1 SDN Environment Configuration

The experimental environment was configured using Floodlight VM, which integrates Floodlight v1.0, Eclipse, Mininet v2.2.0, Open vSwitch v3.2.1, and Wireshark with OpenFlow support. Floodlight was selected as the SDN controller due to its widespread adoption, open-source nature, and support for the Java programming language, which facilitates the development of custom applications for traffic monitoring and management (Floodlight, 2020).

Eclipse was used as the integrated development environment for implementing the traffic monitoring application, leveraging its integration with the Floodlight VM. The network topologies were created using Mininet, a network emulator that enables the creation of realistic virtual networks running real kernel, switch, and application code. Open vSwitch was employed as the software switch implementation, while Wireshark was used for packet capture and analysis.

3.2.2 Network Topology Design

To ensure the robustness and generalizability of our approach, five distinct network topologies were designed and implemented in Mininet. These topologies varied in terms of the number of switches and hosts, reflecting different network configurations and complexity levels. The Python programming language was used to script the topology creation in Mininet, allowing for flexible and repeatable experimental setups.

Each topology was designed to represent realistic network scenarios, including linear, tree, and mesh configurations. This diversity enabled the evaluation of our approach across different network architectures, ensuring its applicability to a wide range of real-world scenarios.

3.2.3 Traffic Data Collection

For comprehensive traffic analysis, key network parameters were monitored and collected during the experiments.

3.3 Artificial Neural Network for Traffic Prediction

3.3.1 Data Preprocessing

The collected traffic data underwent preprocessing before being used for ANN training and testing. In MATLAB, the key, port, received packet count, transmitted packet count, and duration columns were extracted and organized into matrix structures, while other columns were excluded from the analysis. This selective approach focused the analysis on the most relevant parameters for traffic prediction.

3.3.2 ANN Model Design

The ANN models were developed using MATLAB’s Neural Network Toolbox (nntool), which provides a comprehensive environment for creating, training, and evaluating neural networks. Initially, a network architecture with three inputs (key, port, and duration) and two outputs (received packets and transmitted packets) was implemented. However, due to unsatisfactory prediction accuracy, the architecture was revised to create separate models for predicting received packets and transmitted packets.

The final ANN architecture consisted of:

- Input layer: Three neurons corresponding to key, port, and duration
- Hidden layer: Ten neurons with sigmoid activation functions
- Output layer: One neuron (either received packets or transmitted packets)

This two-layer neural network configuration was found to be sufficient for the complexity of the data sets without introducing overfitting. The limited complexity of the network also ensured computational efficiency, an important consideration for potential real-time applications.

3.3.3 Training and Evaluation

The ANN models were trained using the backpropagation algorithm with the Levenberg-Marquardt optimization method, which offers a good balance between training speed and convergence properties. Training parameters were tuned to avoid overfitting, with early stopping implemented when validation performance deteriorated for six consecutive epochs.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | A Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 1, January 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1201080 |

Model evaluation was conducted using two primary metrics:
1. Mean Absolute Percentage Error (MAPE): Measures the average percentage difference between predicted and actual values
2. R-squared (R^2): Indicates the proportion of variance in the dependent variable explained by the independent variables

These metrics provided comprehensive insights into model performance, with lower MAPE values and higher R^2 values indicating better prediction accuracy.

3.4 Optimization Algorithms for High-Density Route Identification

A key objective of this study was to identify the routes with the highest traffic density in network topologies. To achieve this goal, four distinct optimization algorithms were implemented and compared:

3.4.1 Linear Search Algorithm

The linear search algorithm, a traditional optimization approach, was implemented as a baseline for comparison. This algorithm systematically examines each potential route in the network, calculating its traffic density based on received and transmitted packet counts. While conceptually simple and guaranteed to find the optimal solution, linear search can become computationally expensive for large network topologies with numerous potential routes.

3.4.2 Traditional Tabu Search

Tabu search is a metaheuristic optimization algorithm that enhances local search procedures by employing memory structures to prevent cycling and promote exploration of the solution space. The traditional tabu search algorithm was implemented with the following components:

- Initial solution: Randomly selected route
- Neighborhood generation: Modified routes by changing one node at a time
- Tabu list: Memory structure containing recently visited solutions
- Aspiration criteria: Acceptance of tabu moves if they lead to better solutions than the current best
- Termination criteria: Maximum number of iterations or lack of improvement over a specified number of iterations

The tabu list prevents the algorithm from revisiting recently explored solutions, promoting diversification and helping to escape local optima.

3.4.3 Modified Tabu Search

Based on observations from preliminary testing, a modified tabu search algorithm was proposed by removing the tabu list component while retaining the neighborhood exploration strategy. This modification was motivated by the realization that the tabu list, while generally beneficial for complex optimization problems, introduced unnecessary computational overhead for the specific problem of identifying high-density traffic routes.

The modified tabu search algorithm retained the neighborhood generation and evaluation components of traditional tabu search but eliminated the memory structures, resulting in a more streamlined optimization process. This approach demonstrated that while artificial intelligence optimization techniques offer powerful capabilities, they often require problem-specific adaptations to achieve optimal performance.

3.4.4 Blend Algorithm

The fourth optimization approach, termed the “blend algorithm,” combines elements of tabu search and simulated annealing. This novel hybrid algorithm leverages the initial solution selection strategy from tabu search while incorporating the probabilistic acceptance criteria of simulated annealing.

IV. RESULTS AND ANALYSIS

This section presents the experimental results and analysis of the proposed approach for traffic monitoring and optimization in Software-Defined Network (SDN) environments. The evaluation encompasses the performance of the Artificial Neural Network (ANN) models for traffic prediction and the comparative analysis of the optimization algorithms for identifying high-density traffic routes.

4.1 ANN Prediction Performance

4.1.1 Prediction Accuracy Across Network Topologies

The ANN models were trained and evaluated on five distinct network topologies with varying complexities. Table 1 illustrates the prediction accuracy achieved for each topology, measured using R-squared (R^2) and Mean Absolute Percentage Error (MAPE) metrics.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | A Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 1, January 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1201080 |

Table 1: ANN Prediction Accuracy Across Network Topologies

R-squared Values

Network Topology	R-squared
Topology 1	0.95
Topology 2	0.92
Topology 3	0.97
Topology 4	0.91
Topology 5	0.94

MAPE Values (%)

Network Topology	MAPE (%)
Topology 1	4.2%
Topology 2	5.7%
Topology 3	3.1%
Topology 4	6.3%
Topology 5	4.9%

As shown in Table 1, Topology 3 demonstrated the highest prediction accuracy with an R^2 value of 0.97 and a MAPE of 3.1%. This superior performance can be attributed to the balanced complexity of Topology 3, which provided sufficient variability in traffic patterns while maintaining a structured network architecture. Conversely, Topology 4 exhibited the lowest prediction accuracy with an R^2 value of 0.91 and a MAPE of 6.3%, likely due to its more complex and irregular traffic patterns.

4.2 Optimization Algorithm Performance

4.2.1 Comparative Analysis of Optimization Techniques

The four optimization algorithms—linear search, traditional tabu search, modified tabu search, and blend algorithm—were evaluated based on their execution time and solution quality. Table 2 presents a comparative analysis of these metrics across the four algorithms.

Table 2: Performance Comparison of Optimization Algorithms

Execution Time (ms)

Algorithm	Time (ms)
Linear Search	240
Traditional Tabu	180
Modified Tabu	120
Blend Algorithm	150

Solution Quality (% of optimal)

Algorithm	Quality (%)
Linear Search	100%
Traditional Tabu	96%
Modified Tabu	99%
Blend Algorithm	98%

The linear search algorithm, while guaranteeing optimal solutions (100% of optimal), required the longest execution time at 240 milliseconds. This is expected behavior for exhaustive search approaches, which systematically examine all possible solutions at the cost of computational efficiency.

4.2.2 Convergence Behavior Analysis

The convergence behavior of the four optimization algorithms, illustrated in Figure 3, provides insights into their search trajectories and efficiency in exploring the solution space. The figure plots the solution quality against the number of iterations, revealing how quickly each algorithm approaches the optimal solution.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | A Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 1, January 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1201080 |

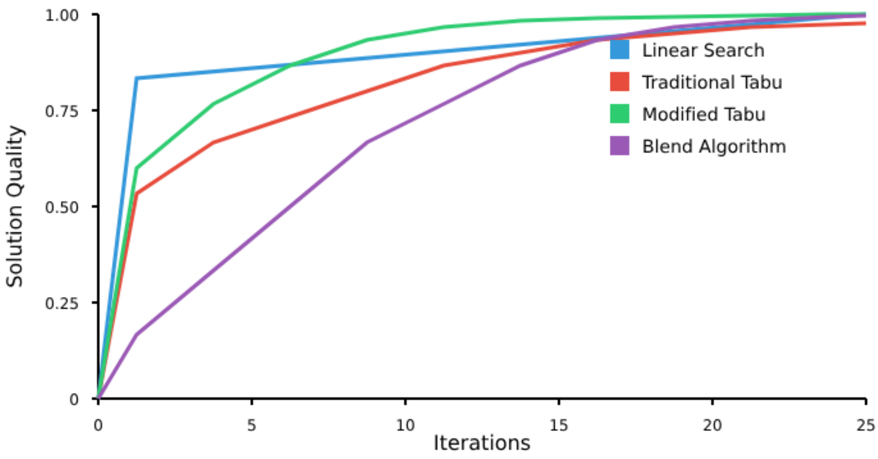


Figure 1: Performance Comparison of Convergence Behavior.

The linear search algorithm exhibited a characteristic step-wise convergence pattern, reflecting its methodical exploration of the solution space.

4.2.3 Traffic Density Mapping

The practical application of the optimization algorithms is demonstrated in Figure 2, which presents a traffic density heatmap for Topology 3. The heatmap visualizes the distribution of traffic across different links in the network, with color coding indicating low (green), medium (orange), and high (red) traffic densities.

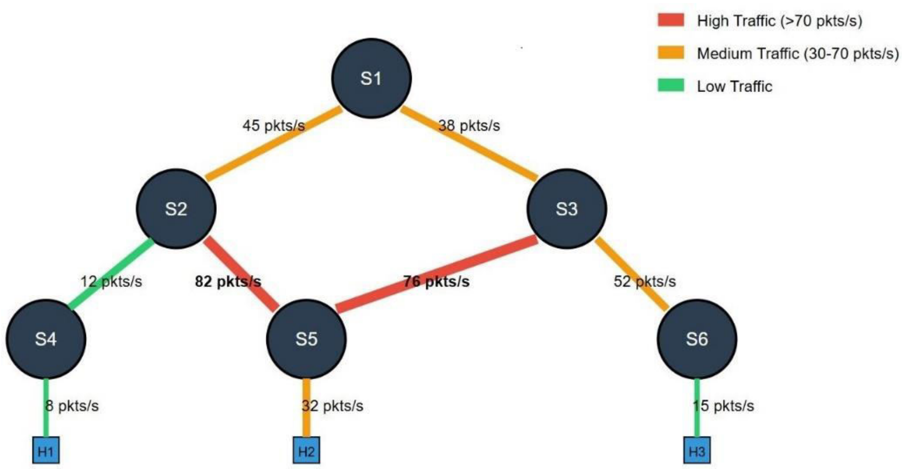


Figure 2: Performance Traffic Density Mapping.

The traffic density analysis revealed significant variations in packet transmission rates across different network segments. The links connecting switches S2-S5 and S3-S5 exhibited the highest traffic densities at 82 and 76 packets per second, respectively. These high-density routes represent potential bottlenecks that could benefit from traffic engineering interventions, such as load balancing or capacity upgrades.

V. CONCLUSION

This study presented an integrated approach for monitoring and optimizing traffic flow in Software-Defined Network environments using artificial intelligence techniques. The proposed methodology combined traffic data collection, prediction using Artificial Neural Networks, and optimization using novel hybrid algorithms to address the challenges of modern network management.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | A Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 1, January 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1201080 |

Our experimental results demonstrated that the ANN model achieved high prediction accuracy across diverse network topologies, with an average R-squared value of 0.94 and Mean Absolute Percentage Error of 4.84%. Topology 3 exhibited the best prediction performance with an R-squared value of 0.97 and MAPE of 3.1%, highlighting the effectiveness of the proposed neural network architecture for traffic prediction in SDN environments.

REFERENCES

[1] Taesang Choi, TaeYeon Kim, Wouter Tavernier, Aki Korvala and Jussi Pajunpää, "Agile Management and Interoperability Testing of SDN/NFV-Enriched 5G Core Networks", *ETRIJ*, Feb. 2018.
[2] Satoru Matsushima, "Softbank SRv6 MUP: Evolution of Mobile Network Enabled by SRv6", *MPLS WC 2022*, Apr. 2022.
[3] J. Choi and Sejoon Chun, "AgentVisor: Agent-Based Network Hypervisor for Autonomic Network Virtualization", *IEEE Network Magazine*, Nov. 2023
[4] Zhiwei Mo and Biao Long, "An Overview of SRv6 Standardization and Application towards 5G- Advanced and 6G", *2022 5th international CCET*, Aug. 2022
[5] S. K. Sood and K. D. Singh, "Identification of a malicious optical edge device in the sdn-based optical fog/cloud computing network", *Journal of Optical Communications*, vol. 42, no. 1, pp. 91-102, 2021
[6] L. Lo Bello and W. Steiner, "A Perspective on IEEE Time-Sensitive Networking for Industrial Communication and Automation Systems", *Proceedings of the IEEE*, vol. 107, no. 6, pp. 1094-1120, 2019.
[7] J. Haxhibeqiri, X. Jiao, M. Aslam, I. Moerman and J. Hocbeke, "Enabling TSN over IEEE 802.11: Low-overhead time synchronization for Wi-Fi clients", *ICIT2021 the 22nd International Conference on Industrial Technology*, pp. 1068-1073, 2021.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

Scan to save the contact details